2017

Improving Syntactic Parsing of Clinical Text Using Domain Knowledge

Min Jiang
University of Texas Health Science Center at Houston School of Biomedical Informatics,
Min.Jiang@uth.tmc.edu

Recommended Citation

Jiang, Min, "Improving Syntactic Parsing of Clinical Text Using Domain Knowledge" (2017). *UT SBMI Dissertations (Open Access)*. 36.
https://digitalcommons.library.tmc.edu/uthshis_dissertations/36

Improving syntactic parsing on clinical corpus using domain knowledge

By

Min Jiang, M.S.

APPROVED:

____
Hua Xu, PhD, Chair

____
Trevor Cohen, PhD

____
Cui Tao, PhD

Date approved: ____________________

Improving syntactic parsing on clinical corpus using domain knowledge

A
Dissertation

Presented to the Faculty of
The University of Texas
Health Science Center at Houston
School of Biomedical Informatics
in Partial Fulfilment of the Requirements for the Degree of

Doctor of Philosophy

By

Min Jiang, M.S.

University of Texas Health Science Center at Houston

2017

Dissertation Committee:

Hua Xu PhD[1], Advisor
Trevor Cohen, PhD[1]
Cui Tao, PhD[1]

[1]The School of Biomedical Informatics

Dedications

I dedicate my dissertation to my family. A special feeling of gratitude to my parents, Tingqi Jiang and Jindao Liu, who encouraged me and helped with housework so I could focus on my research. This dissertation is also dedicated to my loving and supportive wife, Fanhong Meng, and our sweet-hearted little boy, Allan Meng Jiang.

Acknowledgements

First, I would like to thank Professor Hua Xu for his mentorship during the past five years. Without his guidance and persistent help this dissertation would not have been possible. He continually and convincingly conveyed a spirit of rigorousness in regard to research and scholarship. I learned a lot from him on how to conduct research and become a professional researcher.

I also want to thank my other committee members for their advice on my dissertation research. Dr. Trevor Cohen shared with me his thoughts on the syntactic parsing, and Dr. Cui Tao also provided useful comments on the dissertation.

I would like to thank Dr. Heejin Lee and Dr. Yaoyun Zhang, who provided much help with my dissertation. Dr. Lee generated a portion of the codes to extract features using dependency trees in the temporal relation extraction system. Dr. Zhang helped me run the semantic role labeling system using improved parsed result. I could not have finished my dissertation in a timely fashion without them. Additionally, I would like to thank Dr. Yonghui Wu, who provided many comments on the writing and grammar. Moreover, I also would like to thank Dr. Jun Xu, who shared some great ideas with me about the dissertation.

The annotated datasets were obtained from the MiPACQ corpus and 2010 i2b2/VA NLP challenge and I would like to thank Mayo Clinic and the challenge organizers for sharing the datasets.

Abstract

Syntactic parsing is one of the fundamental tasks of Natural Language Processing (NLP). However, few studies have explored syntactic parsing in the medical domain. This dissertation systematically investigated different methods to improve the performance of syntactic parsing of clinical text, including (1) Constructing two clinical treebanks of discharge summaries and progress notes by developing annotation guidelines that handle missing elements in clinical sentences; (2) Retraining four state-of-the-art parsers, including the Stanford parser, Berkeley parser, Charniak parser, and Bikel parser, using clinical treebanks, and comparing their performance to identify better parsing approaches; and (3) Developing new methods to reduce syntactic ambiguity caused by Prepositional Phrase (PP) attachment and coordination using semantic information.

Our evaluation showed that clinical treebanks greatly improved the performance of existing parsers. The Berkeley parser achieved the best F-1 score of 86.39% on the MiPACQ treebank. For PP attachment, our proposed methods improved the accuracies of PP attachment by 2.35% on the MiPACQ corpus and 1.77% on the I2b2 corpus. For coordination, our method achieved a precision of 94.9% and a precision of 90.3% for the MiPACQ and i2b2 corpus, respectively. To further demonstrate the effectiveness of the improved parsing approaches, we applied outputs of our parsers to two external NLP tasks: semantic role labeling and temporal relation extraction. The experimental results showed that performance of both tasks' was improved by using the parse tree information

from our optimized parsers, with an improvement of 3.26% in F-measure for semantic role labelling and an improvement of 1.5% in F-measure for temporal relation extraction.

Vita

2002……………………Bachelor of Science, Computer Science, Jianghan University

2005………………………Master of Science, Software Engineering, Zhejiang University

2008……………………Master of Science, Computer Engineering, University of Florida

2012 to present………..School of Biomedical Informatics, The University of Texas Health Science Center at Houston

Publications

- **Jiang, M.**, Huang, Y., Fan, J. W., Tang, B., Denny, J., & Xu, H. (2015). Parsing clinical text: how good are the state-of-the-art parsers?. BMC Medical Informatics and Decision Making, *15*(1), 1.
- Zhang, Y., Tang, B., **Jiang, M.**, Wang, J., & Xu, H. (2015). Domain adaptation for semantic role labeling of clinical text. Journal of the American Medical Informatics Association, ocu048.
- Tang, B., Feng, Y., Wang, X., Wu, Y., Zhang, Y., **Jiang, M.**, ... & Xu, H. (2015). A comparison of conditional random fields and structured support vector machines for chemical entity recognition in biomedical literature. Journal of Cheminformatics, 7(1), 1.
- Lei, J., Tang, B., Lu, X., Gao, K., **Jiang, M.**, & Xu, H. (2014). A comprehensive study of named entity recognition in Chinese clinical text. Journal of the American Medical Informatics Association, 21(5), 808-814.
- Xu, H., Aldrich, M. C., Chen, Q., Liu, H., Peterson, N. B., Dai, Q., ... & **Jiang, M**. (2014). Validating drug repurposing signals using electronic health records: a case study of metformin associated with reduced cancer mortality. Journal of the American Medical Informatics Association, amiajnl-2014.
- Wei, W.Q., Feng, Q., Jiang, L., Waitara, M.S., Iwuchukwu, O.F., Roden, D.M., **Jiang, M.**, Xu, H., Krauss, R.M., Rotter, J.I. and Nickerson, D.A., 2014. Characterization of statin dose response in electronic medical records. Clinical Pharmacology & Therapeutics, 95(3), pp.331-338.
- Fan, J. W., Yang, E. W**., Jiang, M.**, Prasad, R., Loomis, R. M., Zisook, D. S., ... & Huang, Y. (2013). Syntactic parsing of clinical text: guideline and corpus development with handling ill-formed sentences. Journal of the American Medical Informatics Association, 20(6), 1168-1177.
- Tang, B., Cao, H., Wu, Y., **Jiang, M.**, & Xu, H. (2013). Recognizing clinical entities in hospital discharge summaries using Structural Support Vector Machines with word representation features. BMC Medical Informatics and Decision Making, 13(1), 1.
- Tang, B., Wu, Y., **Jiang, M.**, Chen, Y., Denny, J. C., & Xu, H. (2013). A hybrid system for

temporal information extraction from clinical text. Journal of the American Medical Informatics Association, 20(5), 828-835.

- Ramirez, A.H., Shi, Y., Schildcrout, J.S., Delaney, J.T., Xu, H., Oetjens, M.T., Zuvich, R.L., Basford, M.A., Bowton, E., **Jiang, M**. and Speltz, P., 2012. Predicting warfarin dosage in European–Americans and African–Americans using DNA samples linked to an electronic health record. Pharmacogenomics, 13(4), pp.407-418.
- Feng, Q., Waitara, M. S., Jiang, L., Xu, H., **Jiang, M**., McCarty, C. A., ... & Rieder, M. (2012, March). Dose-response curves extracted from electronic medical records identify sort-1 as a novel genetic predictor of statin potency (ed50). In Clinical Pharmacology & Therapeutics (Vol. 91, pp. S48-S49). 75 Varick St, 9th Flr, New York, NY 10013-1917 USA: Nature Publishing Group.
- Birdwell, K.A., Grady, B., Choi, L., Xu, H., Bian, A., Denny, J.C., **Jiang, M**., Vranic, G., Basford, M., Cowan, J.D. and Richardson, D.M., 2012. Use of a DNA biobank linked to electronic medical records to characterize pharmacogenomic predictors of tacrolimus dose requirement in kidney transplant recipients. Pharmacogenetics and Genomics, 22(1), p.32.
- **Jiang, M**., Chen, Y., Liu, M., Rosenbloom, S. T., Mani, S., Denny, J. C., & Xu, H. (2011). A study of machine-learning-based approaches to extract clinical entities and their assertions from discharge summaries. Journal of the American Medical Informatics Association, 18(5), 601-606.
- Xu, H., **Jiang, M**., Oetjens, M., Bowton, E. A., Ramirez, A. H., Jeff, J. M., ... & Ritchie, M. D. (2011). Facilitating pharmacogenetic studies using electronic health records and natural-language processing: a case study of warfarin. Journal of the American Medical Informatics Association, 18(4), 387-391.
- Tang, B., Chen, Q., Wang, X., Wu, Y., Zhang, Y., **Jiang, M**., ... & Xu, H. (2015). Recognizing disjoint clinical concepts in clinical text using machine learning-based methods. In AMIA Annual Symposium Proceedings (Vol. 2015, p. 1184). American Medical Informatics Association.
- Wu, Y., Xu, J., **Jiang, M**., Zhang, Y., & Xu, H. (2015). A Study of Neural Word Embeddings for Named Entity Recognition in Clinical Text. In AMIA Annual Symposium Proceedings (Vol. 2015, p. 1326). American Medical Informatics Association.
- Xu, J., Zhang, Y., Wang, J., Wu, Y., **Jiang, M**., Soysal, E., & Xu, H. (2015). UTH-CCB: The Participation of the SemEval 2015 Challenge–Task 14. Proceedings of SemEval-2015.
- **Jiang, M**., Wu, Y., Shah, A., Priyanka, P., Denny, J. C., & Xu, H. (2014). Extracting and standardizing medication information in clinical text–the MedEx-UIMA system. AMIA Summits on Translational Science Proceedings, 2014, 37.
- Wu, Y., Tang, B., **Jiang, M**., Moon, S., Denny, J. C., & Xu, H. (2013). Clinical acronym/abbreviation normalization using a hybrid approach. In CLEF (Working Notes).
- Zhang, Y., Cohen, T., Jiang, M., Tang, B., & Xu, H. (2013). Evaluation of vector space models for medical disorders information retrieval. In CLEF (Working Notes).
- **Jiang, M.**, Denny, J. C., Tang, B., Cao, H., & Xu, H. (2012). Extracting semantic lexicons from discharge summaries using machine learning and the C-Value method. American Medical Informatics Association.
- Tang, B., Cao, H., Wu, Y., **Jiang, M**., & Xu, H. (2012, October). Clinical entity recognition using structural support vector machines with rich features. In Proceedings of the ACM Sixth International Workshop On Data And Text Mining In

Biomedical Informatics (pp. 13-20). ACM.

- Liu, M., **Jiang, M**., Kawai, V. K., Stein, C. M., Roden, D. M., Denny, J. C., & Xu, H. (2011). Modeling drug exposure data in electronic medical records: an application to warfarin. In AMIA annual symposium proceedings (Vol. 2011, p. 815). American Medical Informatics Association.
- Xu, H., AbdelRahman, S., **Jiang, M**., Fan, J. W., & Huang, Y. (2011, November). An initial study of full parsing of clinical text using the Stanford Parser. In Bioinformatics and Biomedicine Workshops (BIBMW), 2011 IEEE International Conference on (pp. 607-614). IEEE.
- Xu, H., Lu, Y., **Jiang, M**., Liu, M., Denny, J. C., Dai, Q., & Peterson, N. B. (2010). Mining biomedical literature for terms related to epidemiologic exposures. In AMIA Annual Symposium Proceedings (Vol. 2010, p. 897). American Medical Informatics Association.

Field of Study

Health Informatics

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

Electronic Health Records (EHRs), an important source of observational data, have been widely used to facilitate clinical and translational research. EHRs often contain large amounts of unstructured textual data, which provide valuable information about patients and often serve as a good complement to structured data. To unlock useful information from the clinical narratives, many Natural Language Processing (NLP) methods and tools have been developed [1-4]. Recently, more research has focused on extracting comprehensive information from clinical text (e.g. temporal information of clinical events, relations between clinical events), in addition to simple identification of entities of interest[5-7]. This continues to drive the development of more sophisticated and advanced NLP methods in the medical domain.

Syntactic parsing that generates full syntactic structures of a sentence is a crucial NLP task mediating between linguistic expression and meaning. It is an effective way to generate relations among constituents in the sentence, which can be used for other NLP tasks such as relation extraction. Although syntactic parsing has been widely studied in the open domain, there are very limited studies focusing on parsing clinical text. Moreover, according to sub-language theory[8], clinical text has more restricted semantic patterns compared with general English. Therefore, it may be necessary to leverage clinical domain knowledge to improve syntactic parsing of clinical text.

This dissertation research investigated how to use clinical domain knowledge to improve syntactic parsing performance on clinical corpora. To the best of my knowledge, even though there are many studies working on semantic parsing of clinical corpora[1-3], there is lack of research focussed on improving the performance of syntactic parsing on clinical corpora. In this dissertation, I systematically investigated four aspects of parsing clinical text: 1) Corpora development - I developed annotation guidelines for annotating parse trees of clinical sentences and built two clinical treebanks (Chapter 2); 2) Parser comparison – four state-of-the-art parsers from the open domain were retrained using clinical treebanks and their performance were evaluated carefully (Chapter 3); 3) Ambiguity resolution – semantic information was integrated to revolve syntactic ambiguity from prepositional phrases (PP) and coordinations (Chapter 4); and 4) External validation – to further demonstrate the effectiveness of our parsing approaches, I applied syntactic information generated by our parsers to two external NLP tasks: semantic role labeling and temporal relation extraction from clinical text (Chapter 5).

This chapter provides a review of relevant literature. The recent work of clinical NLP research and the existing work on syntactic parsing in the open and biomedical domains are introduced first. Then, existing research on resolving PP and coordination ambiguities in the open domain is also provided.

1.1 Natural language processing in the medical domain

NLP, as an effective way to convert free text to structure form, has been widely used in the open domain. In the medical domain, many clinical NLP application have been developed to unlock valuable information from clinical text to facilitate clinical studies such as disease phenotypes and patient cohort identification[9, 10], drug repurposing[11] and decision support[12]. Among all the NLP tasks, the most common tasks in the medical domain include Named Entity Recognition (NER), clinical concept encoding and relation extraction.

Named Entity Recognition (NER) is a fundamental task for information extraction. In the medical domain, NER identifies clinical concepts in clinical text and assigns them to pre-defined categories (e.g. disease, medication, lab test, etc.). In the early stages, most of the NER systems in the medical domain used rule-based approaches [1]. However, the maintainability and scalability (human expertise is required to develop and maintain the rule base) of these approaches remain problematic. In 2010, i2b2, a NIH-funded National Center for Biomedical Computing based at Partners HealthCare System, organized a NER challenge to identify three types of named entities including “problem”, “treatment” and “test” on clinical notes from four different institutions. The challenge provided an annotated dataset which contain 837 notes including discharge summaries from Partners Healthcare, Beth Israel Deaconess Medical Center, and the University of Pittsburgh Medical Center and progress notes from the University of Pittsburgh Medical Center. Using this data set, many high-performance machine learning based NER systems have been developed and evaluated [13-15].

However, even once accurately identified, named entities in clinical text are still not easy to use in clinical studies due to high variability of the lexicon of entities. Therefore, some NLP systems, including MedLEE [1, 16], MetaMap [4], cTAKES [2], KnowledgeMap [17], and CLAMP map named entities to concepts in the controlled vocabularies of the Unified Medical Language System (UMLS) . MedLEE, developed by Friedman et al. in the 1990s at Columbia University, is mainly a semantic rule-based system. It was initially designed to extract clinical attributes from radiological reports [18], and then extended to mammography [19], discharge summaries [20, 21] and pathology [22]. MetaMap implemented an algorithm of concept encoding which includes several steps such as partial parsing, variant generation, candidate retrieval, candidate evaluation, and mapping construction. It was developed initially for biomedical literature mining and has recently also been used for clinical note processing. cTAKES combines both rule-based and machine learning techniques under the IBM UIMA framework. The KnowledgeMap, developed by Denny et al. [17, 23] at Vanderbilt University, is another clinical NLP system built to extract clinical concepts with their section headers (by SecTag [24]) and negation status (by NegEx [25]) in documents and map them to UMLS concept unique identifiers. CLAMP is a graphical user interface (GUI) based NLP toolkit recently developed by Xu et al. at the University of Texas Health Science Center at Houston. CLAMP provides multiple existing NLP pipelines that build on a set of high performance NLP components proven in several clinical NLP challenges such as I2b2, ShARe/CLEF, and SemEVAL. Besides, users can easily customize their own pipelines and build annotation for machine learning algorithm by using the CLAMP interface.

Beyond concept normalization, the attributes of a named entity (e.g. assertion, anatomical location of a disease process, temporal information of clinical events, etc.) and the relation between them are also required for most of clinical research. Therefore, many existing NLP systems include such components [2, 26, 27]. For example, for each identified clinical concept, cTAKES also provides the functionality to identify multiple attributes of each identified clinical concept including negation, assertion etc. Aside from the assertion related attributes, CLAMP includes a pipeline to generate clinically related attributes such as body location of disease, lab value of test etc. Moreover, CLAMP also provides a relation extraction module to generate the relation between clinical concepts (e.g. relation between diseases, relation between disease and medication etc.). Extracting relations between entities often requires deeper levels of information about the sentence, such as syntactic structure, to resolve ambiguities in the relations. Therefore, syntactic parsing is the subject of increasing attention from clinical NLP researchers.

1.2 Syntactic parsing in the open domain

Parsing, or formal syntactic analysis, is a fundamental subject in the study of natural language. As an essential prerequisite step to truly understanding the sentential meaning, parsing is performed to represent the structure of a sentence by annotating the relations among its components. There are two major paradigms in syntactic analysis: constituency (phrase structure) parsing and dependency parsing. The paradigms are complementary in terms of expressiveness, and automated methods are available for conversion of one to the other [20]. The constituency paradigm is relatively popular in English, with a huge amount of annotations accumulated over more than a decade (e.g., the Penn Treebank). An example parse in the Treebank style is illustrated in Figure 1, following its latest guideline[28]. The innermost layer of brackets denotes the part-of-speech (POS) tags, which are lexical classes of the tokens and are usually obtained from a separate pre-processor before parsing. On top of the POS tags are the actual phrase-level constituent labels, which represent syntactic roles and imply the semantic scope of each role via the nested bracketing. The rich information in such sentential structure empowers NLP to perform various useful relation-based information extraction and inference tasks.

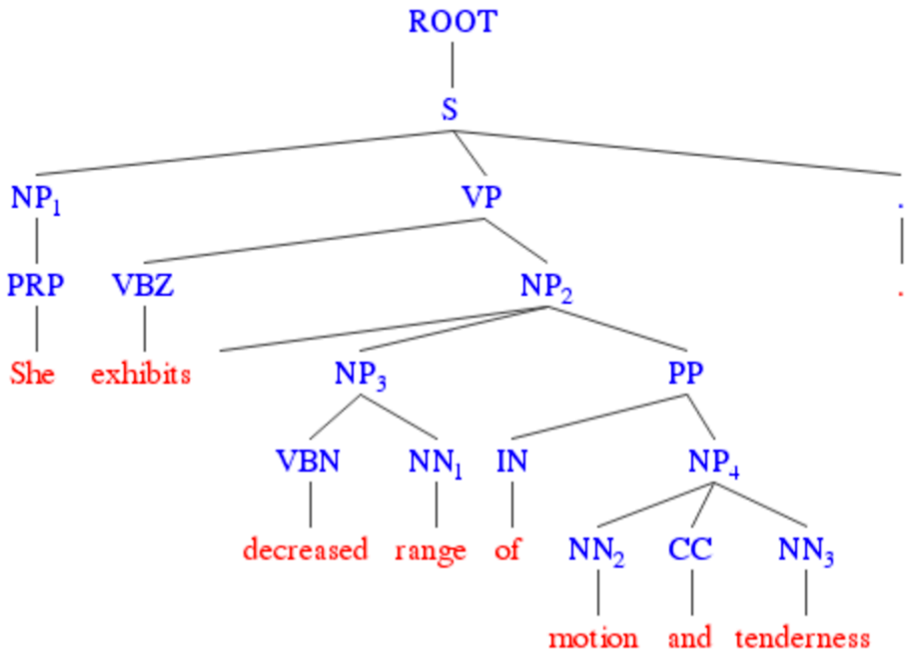


Figure 1. Example of parse tree

Syntactic parsing is a central task in natural language processing because of its importance in mediating between linguistic expression and meaning. Much work has shown the usefulness of syntactic representations for subsequent NLP tasks [29] such as relation extraction, semantic role labeling and paraphrase detection [30]. In the general English domain, early studies of syntactic parsing often relied on symbolic parsing approaches that used manually created deterministic grammars to generate parse trees [31, 32]. Since 1990, statistical approaches have been widely used for syntactic parsing and have shown exceptional performance.

In 1995, Magerman [33] developed one of the first parsers which showed that high-performance parsing could be achieved using only the Treebank based corpora. In his approach, he used the decision-tree learning technique to construct a parse tree of every sentence. In the evaluation, he divided the WSJ corpus from Penn Treebank into 25 sections, numbered 00-24. The parser was trained on sections 02-21 and tested on the 00 section. Moreover, most of subsequent parsers keep using the same data sets in the evaluation when they report the performance on WSJ corpus. The evaluation showed an F-measure of 84.7%. In 1999, Collins [34] demonstrated the use of generative models in syntactic parsing. He extended his probabilistic parser developed in 1996 with three generative models to calculate all probabilities of parse tree head nodes including adjunct/complement distinction and wh-movement. Evaluation showed that these models surpassed Megerman's as well as his own previous parsers and achieved a highest F-measure of 87.8%. In 2004, Bikel [35] used an Expectation-Maximization Model to estimate some feature space parameters in the Collins model. The Bikel parser improved the performance of the Collins' parser and achieved better F-measure on average with all the parameters that he used, demonstrating that his parser was robust and a reliable emulation of the Collins parser. Charniak and Johnson [36] presented a discriminative re-ranking method for constructing high-performance statistical parsers. Based on a coarse-to-fine generative parser, they constructed sets of 50-best parse trees and used them as input into a Maximum Entropy re-ranker, which then selected the best parse. Their parser outperformed all previous generative models and achieved an F-measure of 91.0%. More recently, McClosky et al. [37] presented a two-phase parser that consisted of the Charniak parser and a bootstrapping method for self-training on raw sentences. The McClosky parser boosted performance of the one-phase Charniak parser with an increase of 0.8% in F-measure.

Besides the lexicalized parsers described above, the Stanford parser [38], which was initially developed based on un-lexicalized probabilistic context-free grammar (PCFG) technology, has also shown strong performance and has been widely used across different domains. In addition, Petrov and Klein [39] developed the Berkeley parser, which also implements un-lexicalized technologies by introducing hierarchical coarse-to-fine parsing. It reached an F-measure of 90.1% on the Penn Treebank.

Recently, several studies have focused on using deep learning methods to implement high-performance and efficient syntactic parsers. In 2011, Collobert [40] proposed a new fast discriminative parser, which is based on a recurrent convolutional graph transformer network (GTN). Using only a few basic text features, the parser achieved results comparable to existing state-of-the-art parsers. In 2013, Socher [41] introduced Compositional Vector Grammar (CVG), which combined PCFG with a syntactically united recursive neural network (SU-RNN). The CVG combine the advantage of standard probabilistic context free grammars (PCFG) with those of recursive neural networks (RNNs). PCFG can categorize phrases into specific syntactic categories, while RNN can capture fine-grained syntactic and compositional-semantic information of phrases and words. Furthermore, the CVG approach generalizes the RNN that uses the same weight at all nodes to one with syntactically untied weights, resulting in weights at each node that are conditionally dependent on the categories of the child constituents. As a result, the CVG approach improved the PCFG of the Stanford parser by 3.8% to obtain the F-1 score of 90.4%.

1.3 Syntactic parsing in the biomedical domain

The state-of-the-art parsers have also been applied to the biological domain. For example, Lease and Charniak [42] extended the Charniak parser to process the GENIA corpus [43] generated from MEDLINE abstracts, by leveraging existing domain-specific lexical resources to augment training with the Penn Treebank. More recently, Clegg and Shepherd [44] developed an evaluation method using dependency graphs as an intermediate representation and compared four parsers (Collins parser [34], Bikel parser [35], Stanford parser [38], and Charniak-Lease parser [42]) on the GENIA corpus. Their results showed that the Bikel and Charniak-Lease parsers achieved better performance than the others; but the overall performance of all the parsers dropped when compared to results from the Penn Treebank.

However, few syntactic parsing studies have been done on clinical text from electronic health records (EHRs). Over the past two decades, there has been a growing interest in developing high performance NLP systems for the medical domain. Much detailed patient information is embedded in narratives in EHRs and NLP provides a means to unlock this information for other computerized clinical applications. Early clinical NLP systems such as the Linguistic String Project (LSP)[45, 46] and Medical Language Extraction and Encoding system (MedLEE) [1] are inspired by sublanguage theory, and rely on the relatively restricted semantic constraints of medical language to process text [47]. Despite the success of existing clinical NLP systems on various information extraction tasks [2-4, 17, 45, 46, 48-51], few of them have implemented full syntactic parsing functionality. Some studies extended the general English parsers such as the

Stanford Parser using a medical lexicon for clinical text processing [52], but no formal evaluation of syntactic parsing has been done for these parsers. Fortunately, recent initiatives in the clinical NLP community have led to generation of detailed annotation guidelines, as well as richly annotated corpora. For example, the MiPACQ corpus, which contains pathology and other clinical notes from the Mayo Clinic, has multiple layers of annotations, including named entities, syntactic parse trees, dependency parse trees, and semantic role labeling on 13,091 sentences [53]. It was used to retrain a dependency parser and achieved a highest labelled attachment score of 0.836. Additionally, in 2015, Yan [54] extended Stanford parser's grammar with the SPECIALIST lexicon and statistics collected from an operative notes corpus to achieve the F-score of 89.90% of syntactic parsing on operation notes.

1.4 Resolving ambiguity of Prepositional Phrase attachment

The error of prepositional phrase (PP) attachment is one of the common issues in syntactic parsing. It refers to the problem of determining the correct attachment site for a PP, conventionally in structures such as "V NP PP" [55, 56]. For instance, in the sentence "I ate a pizza with a fork", the PP "with a fork" could attach either to the verb "ate" or to the noun phrase "a pizza". In this case, the verb is the correct attachment site (as the fork is involved in the act of eating). For the sentence "I ate a pizza with tomato sauce", on the other hand, the noun phrase ("a pizza") is the correct attachment site (as the tomato sauce is applied to the pizza). PP attachment is a structural ambiguity problem, which is a common issue when parsing a sentence.

In the general domain, many studies have been conducted to resolve the PP attachment ambiguity and they generally can be divided into two categories: supervised vs.

unsupervised approaches. Supervised approaches leverage thesauri to group words into semantic classes, to address the problem of data sparseness. Brill and Resnik (1994) [56] developed a supervised transformation-based learning method with lexical and conceptual classes derived from Word-Net, achieving a precision of 82% on 500 randomly selected ambiguous prepositional phrases examples. Ratnaparkhi et al. [56] created a benchmark dataset of 27,937 quadruples (v,n1,p,n2), extracted from the Wall Street Journal. Based on this dataset, they trained a maximum entropy model and a binary hierarchy of word classes derived by mutual information, achieving a precision of 81.6%. Collins and Brooks (1995) [56] used a supervised back-off model and reported a precision of 84.5% on the Ratnaparkhi test set. Stetina and Makoto (1997) [57] used a supervised method with a decision tree and WordNet classes to achieve a precision of 88.1% on the same test set. Toutanova et al. [58] adopted a supervised method that makes use of morphological and syntactic analysis and WordNet synsets, yielding 87.5% accuracy.

For unsupervised approaches, the attachment decision depends largely on co-occurrence statistics drawn from text collections. The pioneering work in this area was conducted by Hindle and Rooth (1993) [59]. Using a partially parsed corpus, they calculated and compared lexical associations over subsets of the tuple (v, n1, p), ignoring n2, and achieved 80% precision and 80% recall. Ratnaparkhi [60] developed an unsupervised method that collected statistics from text annotated with part-of-speech tags and morphological base forms and they achieve a precision of 81.9% on the Ratnaparkhi test set. Pantel and Lin [60] described an unsupervised method that used a collocation database, a thesaurus, a dependency parser, and a large corpus (125M words), achieving a precision of 84.3% on the Ratnaparkhi test set. Using simple combinations of web-based n-grams, Lapata and Keller [60] achieved less impressive results, with precision in the low 70's range.

Even though there are some studies aiming to resolve PP attachment ambiguities in the open domain, there is a lack of studies on resolving PP attachment in clinical text.

1.5 Resolving ambiguity of coordination

Coordination error is another common issue for syntactic parsing. Coordination is a procedure that links two sentence elements called conjuncts, and it is very common in language. Conjoined structures may be globally or temporarily ambiguous because it is grammatically permissible to conjoin any type of constituent as long as the conjuncts are from the same syntactic category. For example, the sentence of "She is not having any incontinence or suggestion of infection at this time." can be interpreted into two ways: 1) "incontinence" and "suggestion" are conjoined phrases; 2) "incontinence" and "suggestion of infection" are conjoined phrases.

In the general English domain, most previous attempts to resolve coordination ambiguity have focused on a particular type of NP coordination. Both Resnik [61] and Nakov and Hearst [62] considered NP coordinations of the form "n1 and n2 n3" where two structural analyses are possible: ((n1 and n2) n3) and ((n1) and (n2 n3)). To resolve such ambiguity, Resnik combined number agreement information of candidate conjoined nouns, an information theoretic measure of semantic similarity, and a measure of the appropriateness of noun-noun modification. Nakov and Hearst's [62] disambiguation method is to combine Web-based statistics on headword co-occurrences with other mainly heuristic information sources. A probabilistic approach was presented in (Goldberg, 1999)[63], where an unsupervised maximum entropy statistical model was used to disambiguate coordinate noun phrases of the form n1 preposition n2 cc n3.

In the medical domain, as with PP attachment, there is no prior research work focused on resolving coordination ambiguities in clinical text.

1.6 Summary

In this chapter, relevant literature focussed on syntactic parsing in both open domain and the medical domain was reviewed. As discussed in the above sections, syntactic parsing is a critical task of NLP and has been extensively studied in open domain. However, limited work has been done regarding the methods of syntactic parsing in the medical domain. Moreover, few annotated clinical corpora of parse trees are available for developing and evaluating syntactic parsing methods in the medical domain. Therefore, there is a need for further research to create clinical treebanks and develop customized syntactic parsing approaches for clinical text, with the aim to improve other NLP tasks using extracted syntactic information. My hypothesis is that by leveraging clinical corpora and domain specific semantic knowledge, existing open-domain parsers' performance on processing clinical text can be improved. To validate the hypothesis, I propose the following specific aims:

Aim 1 – Annotation of clinical treebanks

Aim 2 – Evaluation and extension of the state-of-the-art parsers using clinical treebanks

Aim 3 – Using semantic information to reduce PP attachment and coordination ambiguities in the parsed result

Aim 4 - Validation of the effectiveness of the optimized parsers in clinical NLP tasks

Chapter 2. Develop treebanks of the clinical text

2.1 Introduction

One of the main reasons for the paucity of studies on syntactic parsing in the medical domain is the resource-intensive nature of establishing an infrastructure to support the development of clinical text parsers. Two challenges in particular are:

1) Limited availability of raw and annotated clinical texts and standards for annotation (Chapman [64] et al.), with the downstream effect of limiting reproducibility and collaboration. In contrast, general English parsers have achieved accuracies in the high 80's, benefiting from abundantly annotated data and well-documented guidelines.

For example, the Penn Treebank project [65] syntactically annotated sentences in the over one million word Wall Street Journal (WSJ) corpus and released the annotation guidelines to the public. Both the corpus and the guidelines have subsequently served as the basis for an enormous amount of research in parser development as well as annotation of new corpora.

The clinical NLP community, while being aware of the benefits of high-accuracy parsers and moving generally in this direction, has focused mostly on semantic or even higher level annotation for specific applications [66-68]. One formal initiative that addresses this issue is the Strategic Health IT Advanced Research Projects (SHARP) by Mayo Clinic and its collaborating institutions[69], which have developed a MiPACQ (Multi-source Integrated Platform for Answering Clinical Questions) clinical corpus with layered annotations including syntactic parse trees and shared their guidelines [53, 70]. However, there does not appear to be any existing approach that has adequately addressed the challenges in syntactic representation for ill-formed sentences, which are especially common in clinical text [64].

2) The intrinsic properties of medical sublanguage [47, 70] make linguistic annotation extremely knowledge-intensive, therefore, placing a high bar on the annotation. Many clinical sentences are telegraphic, with missing subjects and/or verbs. As a result, these sentences make interpretation difficult due to the frequent ambiguities found in clinical text and pose a challenge to existing parsers that are modelled on well-formed sentences. Filling in the gaps in such cases requires appropriate domain knowledge, without which the process of analysing the syntactic structure and interpreting the meaning would be highly error-prone. Therefore, it is difficult to apply a general annotation guideline to clinical corpus annotation.

Based on the challenges summarized above, it is clear that it is extremely valuable to develop syntactically annotated corpora for the medical domain, by following linguistically sound annotation guidelines. Ideally, the annotations/guidelines should be compatible with some commonly accepted style of syntactic annotation for general English (e.g., that of the Penn Treebank) so that a great number of existing methods/tools can be reused. At the same time, it should also address unique characteristics of the medical language itself.

2.2 Methods

2.2.1 Develop a parse tree annotation guideline using progress notes

2.2.1.1 Data selection and preprocess

First, we used progress notes from the University of Pittsburgh Medical Center (UPMC) distributed in the 2010 i2b2/VA Clinical NLP Challenge. A physician manually reviewed the progress notes and selected 25 of them by identifying those concerning general medicine and excluding those that apparently contained copy-pasted redundant information. To eliminate confounding errors in steps prior to parsing, a linguist performed manual tokenization and POS tagging for the 25 notes, consulting a physician on cases that required domain knowledge. Based on the gold standard tokenization, we automatically sampled sentences with at least 3 tokens of length for this study. Before distributing the notes for manual parsing, we pre-parsed them using the Stanford Parser with the general English PCFG model that was provided in the official package.

2.2.1.2 Annotation guideline and progress notes Treebank development

Four major stages were included in developing our annotation guidelines along with iterative refining of the deliverable corpus (see Figure 2 for flow chart):

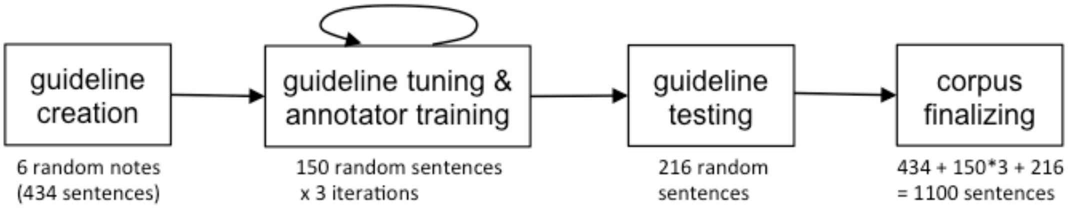

guideline creation
6 random notes (434 sentences)
guideline tuning & annotator training
150 random sentences x 3 iterations
guideline testing
216 random sentences
corpus finalizing
434 + 150*3 + 216 = 1100 sentences

Figure 2. Stages of developing annotation guidelines

Based on the manual analysis of institution-specific notes other than the 25 notes selected for this study, a linguist drafted a preliminary set of clinical parsing guidelines adapted from the Penn Treebank [65]. We applied the error handling strategy introduced by Foster [71] and found that specifically inserting null elements for ignored (or missing) words was helpful in restoring the proper syntax of ungrammatical clinical sentences. The guideline creation was not only knowledge-driven but also involved annotating real clinical text. We used the application WordFreak [72] for performing our manual annotation. There were two teams involved in the development of the annotation guideline, including one from Kaiser Permanente and one from Vanderbilt University . In the initial batch, both teams annotated 6 of the 25 notes (with consulting physicians for domain-specific interpretation) and discussed/reconciled any disagreements. After each round, the draft guidelines were revised accordingly to address the issues discovered.

In the second stage, we performed three more rounds of annotation to train two human annotators and to fine-tune the guidelines. For each round, we randomly sampled 150 sentences from the remaining 19 notes (not used in the first stage) and had both the linguist and a non-linguist annotate them by following the guidelines and consulting physicians whenever needed. The Evalb program (the version that came with the Stanford Parser) was used to compute the inter-annotator agreement rate based on the F-1 measure. The annotators met after assessing the agreement rate to reconcile and discuss applicability of the guidelines. These three rounds with 450 sentences were meant to facilitate convergence among the annotators and the guidelines (by revising any instructions that were considered problematic).

To test the learnability/stability of our guidelines, we randomly sampled another 216 sentences (intended to round up the deliverable corpus to 1,100 sentences) from the 19 notes and had both annotators annotate them. An agreement rate was computed to evaluate whether the annotators could generate consistent annotations by following the latest guidelines. Additionally, to investigate the potential bias introduced by pre-parsing with the Stanford Parser, we had the linguist annotate two of the six stage-one notes (interrupted with a wash-out period of a half year) directly from the POS-tagged sentences. The linguist's self-agreement rate (between her annotations using the pre-parsed vs. those using only the POS-tagged) was computed to evaluate whether the consistency can be achieved with/without the factor of pre-parsing.

After the agreement rates were assessed to be satisfactory in the last round, the annotators met to discuss/resolve any remaining disagreements and then the linguist applied the latest guidelines in preparing a final version of the deliverable corpus.

2.2.2. Extend the guideline to create a Treebank of discharge summaries

We applied the guideline to annotating discharge summaries contained in the 2010 I2b2 clinical NLP challenge, a total of 237 documents. Every sentence in these discharge summaries was pre-processed by the Stanford parser and a researcher in clinical NLP manually reviewed each parse tree generated by the Stanford parser and corrected it based on the annotation guideline. When there was any question about a parse tree, a linguist and a physician were consulted for correct annotation.

2.2 Results

2.2.1 Annotation guideline development

Inter-annotator agreement rates were computed to measure the consistency between institutions in following the guidelines. Table 1 shows the progressive rates from the three iterations of guideline tuning to the final independent testing. We can see the agreement rates climbed steadily over the tuning iterations and culminated at 0.930 in the final testing phase. The intra-annotator agreement rate of the linguist in annotating from Stanford Parser pre-parsed sentences versus from POS-tagged sentences only was 0.948, with 0.791 perfectly self-agreed parse trees. These results indicate that the annotators were able to perform syntactic parsing with acceptable consistency by following the guidelines.

Table 1. Inter-annotator agreement rates in guideline tuning and final testing

	Tuning 1	**Tuning 2**	**Tuning 3**	**Final testing**
Number of sentences	150	150	150	216
Agreement rate	0.872	0.887	0.903	0.930
Proportion of perfectly agreed upon parse trees	0.633	0.660	0.693	0.713

Our annotation guidelines are based on the original Bracketing Guidelines for Treebank II Style [12], with modifications to accommodate the properties of medical sublanguage. Several noteworthy adaptations are summarized in the following:

a) Insert missing elements: We adopted Foster's[71] approach in annotating sentences with omitted words. We insert a null element 0-NONE- for the inferred missing word to restore the intended syntactic structure.

b) Mark superfluous and redundant elements: Superfluous and redundant elements that cannot be accommodated in a sentence or phrase structure are annotated with the constituent “X”, including some punctuation marks, symbols, and words.

c) Handle symbols with inferable syntactic roles: “Header: value” expressions occur frequently in clinical notes. Some of them can be legitimately annotated as complete sentences if the colon is interpreted as a verb. In such cases, the colon is allowed to precede a verb phrase while its POS tag should still remain “:”.

d) Interpret syntactic roles of Latin abbreviations: instead of using FW (foreign word) for Latin abbreviations, we try to infer their functioning POS tags and syntactic roles.

e) Respect domain-specific semantic structure of complex phrases. We try to accurately represent the internal semantic structure of medical expressions in annotating the constituents. When there are alternative grammatical parses, the one that better captures the intended meaning is preferred.

2.2.2 Annotation corpus

The annotated corpus contains 1,100 sentences, with a median length of 8 tokens per sentence. Table 2 shows the distribution of syntactic constructs (constituents) in our annotated corpus, aligned side by side with that in an arbitrarily selected WSJ sub-corpus (the 00 section) of 1,921 sentences. Table 3 shows a list of those rules that involve restoration of missing elements by inserting a 0-NONE- node. For example, we can see the most frequent rule involving element restoration is VP → -NONE- NP.

Table 2. Constituent distribution in the annotated corpus with comparison to a general corpus

Constituent label	**Constituent description**	**% in our annotated clinical corpus**	**% in a subset of WSJ corpus**
NP*	Noun phrase	40.00	42.37

VP*	Verb phrase	16.85	19.68
S	Sentence	12.18	12.93
PP*	Prepositional phrase	8.56	12.74
FRAG*	Fragment	7.65	0.21
ADJP*	Adjective phrase	4.51	1.66
ADVP	Adverb phrase	2.87	2.57
NX*	Head in complex NP	2.14	0.26
SBAR*	Clause introduced by a subordinating conjunction	1.42	3.93
PRN*	Parenthetical	1.41	0.50
LST*	List marker	0.78	0.03
X*	Unknown, uncertain, or unbracketable constituent	0.58	0.01
WHNP*	Wh-noun phrase	0.44	0.99
PRT*	Particle	0.21	0.35
UCP*	Unlike-coordinated phrase	0.17	0.04
WHADVP	Wh-adverb phrase	0.14	0.25
QP*	Quantifier phrase	0.06	0.91
CONJP	Conjunction phrase	0.02	0.04
WHPP	Wh-prepositional phrase	0.01	0.05
SQ	Inverted yes/no question, or main clause of a wh-question	0	0.04
SINV*	Inverted declarative sentence	0	0.30
SBARQ	Direct question introduced by a wh-word or wh-phrase	0	0.03
RRC	Reduced relative clause	0	0.02
NAC*	Not a constituent	0	0.08
INTJ	Interjection	0	0.01

Table 3. Grammar rules involving restoration of missing elements in our annotated corpus

Frequency	**Grammar rule**
159	VP → -NONE- NP
52	NP → -NONE-
41	VP → -NONE- VP
25	PP → -NONE- NP
21	VP → -NONE- ADJP
6	VP → -NONE- PP
3	VP → -NONE- NP PP
2	NP → JJ -NONE-

2	VP → -NONE- SYM NP
1	VP → -NONE- ADVP VP
1	VP → -NONE- NP PRN
1	VP → -NONE- NP , NP , NP
1	PP → -NONE- NP ADVP
1	VP → -NONE- NP NP
1	ADJP → ADVP -NONE-

2.3 Discussion

The main purpose of this study was to develop/evaluate/share domain-customized parsing guidelines along with a real clinical corpus annotated accordingly. The promising inter-annotator agreement rate (0.930) indicated reliability of the guidelines, and the accuracy (0.811) of a statistical parser retrained with the corpus demonstrated reasonable usability of the annotations. To our knowledge, the current work was the first to introduce Foster's error-handling approach to ill-formed clinical sentences. We do not claim it to be the most suitable and final solution to annotating ungrammatical clinical sentences. Rather, the rationale was merely to perform solid experiments and share the results. It is hoped that the community will gradually converge to a common consensus by combining the advantages of different proposals.

As mentioned in the Background section, the study was partly motivated towards addressing the limited interoperability in medical text parsers that involve proprietary semantic grammar. We believe our sharing of the standard-conforming corpus is critical in a data-driven, sustainable model that attracts constant pursuit of questions/solutions for research and development. However, it should be emphasized that a general syntactically annotated corpus is not contradictory to the value of any existing semantic approaches. If combined appropriately, syntax and semantics can in fact complement each other in forming robust parsing solution to clinical narratives. Specifically, when superimposed with semantic annotations on the same corpus, the Treebank constituents can facilitate automated derivation of a semantic grammar in flexible ways.

As by-products, the study yielded interesting findings as well as research questions:

1) The simple comparison of constituent distribution between the annotated corpus and a WSJ subset served as proof of concept that clinical text does differ syntactically from non-clinical English. However, it is an open question whether the progress note sample in this study is representative of clinical text. In other words, could syntactic composition differ considerably even among different clinical genres and also between clinical text that was dictated and versus typed?

2) The number of iterations and amount of training notes provided a hint on the effort required to achieve reasonable annotation consistency. Our results suggest it would take at least three iterations of annotating/adjudicating on more than 500 sentences in total for the annotators to reach a higher than 0.9 agreement rate. However, there is a need for a larger scale comparative study to verify the generality of our findings.

3) Combining clinical text with a certain amount of Treebank training sentences resulted in the most accurate parser model. A purely general English model achieved an accuracy of only 0.656, but the mixture boosted the purely clinical parser model's accuracy from 0.769 to 0.811. One hypothesis is that the size of our corpus (1100 sentences) is still not sufficient to train a statistically robust parser, and therefore even off-domain annotations

can help with a smoothing effect. The research question here is: How large should the clinical corpus be in order to independently train a parser? And before the sufficiency of domain-specific training data is achieved, how can we reliably estimate the optimal ratio to mix the heterogeneous corpora?

2.4 Conclusion

With an iterative approach, we developed syntactic parsing guidelines for clinical text and annotated a set of 1100 sentences in progress notes accordingly. The guidelines are compatible with the standard Penn Treebank syntactic annotation style and include special adaptations to accommodate clinical sublanguage properties. Two annotators (a linguist and a computer scientist) reached an agreement rate of 0.930 in the final independent evaluation, which indicates consistency in following the guidelines. As simple validation of usefulness, retraining a statistical parser with the annotated corpus achieved a best accuracy of 0.811 (by involving also some off-domain training sentences).

Chapter 3. Leveraging clinical treebanks to improve state-of-the-art parsers performance on clinical corpora

In the previous chapter, we developed an annotation guideline for clinical treebank and built two clinical corpora including a discharge summary treebank and a progress note treebank. Both Treebanks can be used to re-train existing parsers to recognize syntactic structures of sentences in clinical text. In this chapter, we re-trained several state-of-the-art parsers in the open domain and evaluated their performance on different types of clinical notes, by using Treebanks developed by us and others.

3.1 Introduction

In the general domain, there are several state-of-the-art parsers that have achieved great performance in English corpora such as the Penn Treebank etc.

3.1.1 The Stanford parser

In 2003, Klein proposed an un-lexicalized parser, the Stanford parser. It achieved the bracket F-measure of 86.36%, which was better than the early lexicalized PCFG models. In the study, Klein described several simple, linguistically motivated annotations on both non-terminal and POS tagging level, which do much to close the gap between basic PCFG and state-of-the-art lexicalized models. In 2013, Socher proposed a deep learning based solution for Stanford parsers, which used a syntactically united recurrent neural network (SU-RNN) to combine PCFG generated from the treebank and semantic information from the pre-trained word embeddings. As a result, it increased the F-measure to 90.4%.

3.1.2 The Bikel parser

In 1997, Collins proposed a new statistical parsing model, which is a generative model of lexicalized context-free grammar. Then he extended the model to include a probabilistic treatment of both sub-categorization and wh-movement. The model achieved the 88.1/87.5% constituent precision/recall[34]. In 2004, Klein implemented the Bikel parser, which is based on the Collins parsing model. Additionally, Klein implemented a flexible constraint-satisfaction mechanism to build the model for unknown events in the training data. It used the Expectation-Maximization algorithm to estimate the feature space parameters in the Collins model [35].

3.1.3 The Charniak parser

In 2000, Charniak presented a new parser based on "maximum-entropy-inspired" model for conditioning and smoothing, which achieved a F-score of 90.1% for sentences of length 40 words and less, and a F-score of 89.5% for sentences of length 100 words and less. In 2005, Charniak used a simple yet novel method for constructing sets of 50-best parse trees based on a coarse-to-fine generative parser, then those 50-best parse trees were fed into a discriminative re-ranker which is based on Maximum Entropy model to select the best parse tree. As a result, it achieved an F-score of 91.0% on sentences of length 100 words or less [36].

3.1.4 The Berkeley parser

In 2007, Petrov introduced the Berkeley parser, which made several improvements to unlexicalized parsing with hierarchically state-split PCFGs. First, he presented a novel coarse-to-fine method in which a grammar's own hierarchical projections are used for incremental pruning, including a method for efficiently computing projections of a grammar without a Treebank. Second, he compared various inference procedures for state-split PCFGs from the standpoint of risk minimization, paying particular attention to their practical tradeoffs. As a result, the Berkeley parser achieved F-score of 90.0% on an English corpus [39].

3.1.5 Application of parsing in the medical domain

Recently, the research community has started applying existing parsing algorithms to clinical text. In 2011, Xu randomly selected 50 sentences in the clinical corpus from the 2010 i2b2 NLP challenge and manually annotated them to create a gold standard of parse trees. In the study, the evaluation showed that the original Stanford Parser achieved a bracketing F-measure (BF) of 77% on the gold standard. Further, the study assessed the effect of part-of-speech (POS) tags on parsing and the results showed that manually corrected POS tags achieved a maximum BF of 81%. Also, it analysed parsing errors from the Stanford Parser and provided valuable insights for large-scale parse tree annotation for clinical text. In 2013, Daniel reported the performance of the OpenNLP constituency parser on a corpus combining general domain data and the MiPACQ data described in this manuscript., The parser achieved a labeled $F1$ score of 0.81 on a corpus consisting of clinical and pathology notes when tested on held-out data of clinical and pathology notes [53]. In 2015, Yan expanded Stanford parser's grammar using the SPECIALIST lexicon and reported its performance on operative notes [54].

3.2. Methods

3.2.1 Choose state-of-the-art parsers

Basically we wanted to follow Clegg and Shepherd's study [44], which compared four parsers. However, we excluded the Collins parser because it lacks a simple way to re-train the parser using a different corpus. Furthermore, the Berkeley parser, as a very popular parser in the English domain, has good potential for high performance on the cross domain. As a result, we included four parsers in the study: the Stanford parser [38], the Bikel parser [35], the Charniak parser [36] and the Berkeley parser [39]. In addition, we also included a compositional vector grammar based parser [73] as a deep learning based Stanford parser.

3.2.2 Choose clinical treebanks

We evaluated the performance of existing parsers on three clinical treebanks including two that was built previously: 1> The ProgressNotes treebank originated from the clinical notes from the 2010 I2b2 NLP challenge; 2> The DischargeSummaries treebank and the MiPACQ treebank, which consists of annotated clinical and pathology notes related to colon cancer from Mayo Clinics. When retraining our developed treebank, we removed the annotation for missing elements in this experiment.

3.2.3 Strategy of parsing

The parsing experiment included three steps: firstly, we ran four State-of-the-Art parsers with the default settings. Then we used 5-fold cross validation mechanisms to retrain the parsers on both treebanks. Finally, to test if combining treebank from clinical and other domain could achieve better results, we ran the parsers with each treebank combining the WSJ corpus from Penn Treebank respectively. For the deep learning based Stanford parser, when retraining on clinical corpus, we generated new word embedding vectors from clinical text in the MIMIC-III dataset[74] using continuous bag-of-words (CBOW) (context window size of 5, threshold for downsampling the frequent words of 1e-3) in Word2vec model and fed them into the deep neural network.

3.3 Results

Table 4 shows the experimental results on the ProgressNotes treebank. The deep learning based Stanford parser achieved the best performance of 71.14% BF, with the default settings. Compared to the default setting, re-training on the clinical treebank improved the performance for three parsers, with the biggest boost achieved by the deep learning based Stanford parser (from a F-score of 71.14% to 78.15%). When the combined corpora of both progress notes and WSJ articles were used for training, the BF of the Berkeley parser and Charniak parser increased by 3.9% (from 71.87% to 75.77%) and 3.52% (from 70.01% to 73.53%); however, both the Stanford and the Bikel parsers dropped slightly in their performance.

Table 5 shows the results obtained using the DischargeSummaries treebank. With the default setting, the Stanford parser again achieved the best performance among all the parsers. Upon re-training on the MiPACQ Treebank alone, all the four parsers had a big leap in performance with the Berkeley parser showing the maximum increase (increased from a BF of 69.55% to 86.39%). Re-training on the combined treebanks of MiPACQ and WSJ led to marginal increases in performance for all the parsers. Among all the parsers, the Berkeley parser achieved the best BF of 86.05% when both MiPACQ and WSJ treebanks were used.

Table 6 shows the results obtained using the MiPACQ treebank. With the default setting, the Berkeley parser again achieved the best performance among all the parsers. Upon re-training on the MiPACQ treebank alone, all the four parsers had a big leap in performance with the Stanford parser showing the maximum increase (increased from a BF of 69.55% to 86.39%). Re-training on the combined treebanks of MiPACQ and WSJ led to marginal increases in performance for all the parsers. Among all the parsers, the

Berkeley parser achieved the best BF of 86.05% when both MiPACQ and WSJ treebanks were used.

Table 4. Results for four parsers on the ProgressNotes treebank

Parser	Training corpus	BR (%)	BP (%)	BF (%)
Stanford	Default	70.31	70.27	70.29
	Clinical	76.22	71.31	73.68
	Clinical + WSJ	74.27	71.16	72.68
Stanford (deep learning)	Default	70.61	71.68	71.14
	Clinical	79.91	76.47	78.15
Bikel	Default	64.20	69.20	66.60
	Clinical	71.85	73.05	72.45
	Clinical + WSJ	70.85	73.92	72.35
Charniak	Default	62.91	75.03	68.44
	Clinical	65.82	74.78	70.01
	Clinical + WSJ	75.89	71.31	73.53
Berkeley	Default	66.63	65.08	65.85
	Clinical	77.24	67.19	71.87
	Clinical + WSJ	79.41	72.46	75.77

Table 5. Results for four parsers on the DischargeSummaries treebank

Parser	Training corpus	BR (%)	BP (%)	BF (%)
Stanford	Default	68.37	72.06	70.17
	Clinical	83.10	82.52	82.81
	Clinical + WSJ	77.76	80.57	79.14
Stanford (deep learning)	Default	68.50	69.78	69.13
	Clinical	84.07	83.58	83.82
Bikel	Default	65.14	72.48	68.61
	Clinical	76.32	77.86	77.08
	Clinical + WSJ	74.96	77.76	76.34
Charniak	Default	62.14	75.75	68.27
	Clinical	74.57	76.63	75.58
	Clinical + WSJ	74.56	83.15	78.62
Berkeley	Default	67.33	74.68	70.82
	Clinical	85.03	83.89	84.45
	Clinical + WSJ	82.41	84.56	83.47

Table 6. Results for four parsers on the MiPACQ treebank

Parser	Training corpus	BR (%)	BP (%)	BF (%)

Stanford	Default	75.54	74.41	74.97
	Clinical	84.35	85.45	84.90
	Clinical + WSJ	84.89	85.24	85.06
Stanford (deep learning)	Default	75.78	73.45	74.59
	Clinical	85.83	84.80	85.31
Bikel	Default	73.49	75.78	74.62
	Clinical	77.59	78.09	77.84
	Clinical + WSJ	77.43	78.63	78.03
Charniak	Default	70.63	78.11	74.18
	Clinical	80.88	86.39	83.54
	Clinical + WSJ	80.65	86.76	83.59
Berkeley	Default	66.30	73.14	69.55
	Clinical	85.94	86.85	86.39
	Clinical + WSJ	86.03	86.08	86.05

3.4 Discussion

Full syntactic parsing is an important area of clinical NLP research, but it has not been extensively explored so far. In this study, we conducted the first formal evaluation to compare the performance of four state-of-the-art English parsers on clinical notes using three clinical treebanks. When all three treebanks were retrained on the parsers, the highest average BFs of 86.39%, 84.45% and 78.15% were achieved by the Berkeley parser for the MiPACQ and DischargeSummaries corpus and the deep learning based Stanford parser for the ProgressNotes treebank respectively.

As expected, existing parsers achieved lower performance on clinical text than previously reported results on general English text, when they were directly applied to clinical text. For instance, on the MiPACQ corpus, the Stanford parser showed a decrease of 11.35% in BF (from 86.32% to 74.97% in this study). When the existing parsers were re-trained on the clinical treebanks, their performance increased. For the progress notes treebank, there were 3.39%, 7.01%, 5.85%, 1.57% and 6.02% increases in BF for the Stanford, deep learning based Stanford, Bikel, Charniak and Berkeley parser, respectively. For the

MiPACQ corpus, the increases were 8.93%, 10.72%, 3.22%, 9.36% and 16.84%, which were much higher than increases in the progress notes corpus, probably due to the larger sample size of the MiPACQ corpus (about 10 times larger than the progress notes corpus – 10,661 vs. 1,025 sentences). These findings suggest that re-training on clinical corpora is necessary for developing high-performance statistics-based parsers for clinical text. It also indicates the need for building annotated clinical treebanks.

Although there is growing interest in building annotated clinical corpora, the sizes of these corpora are often limited due to the high cost of physician annotators. Large-scale corpora from other domains, such as the Penn Treebank, are available and should be leveraged for clinical parsing. That is the motivation of the combination approach proposed in this study. For progress notes, direct combination of the WSJ corpus and the clinical corpus showed varying results among the four parsers. It largely improved the performance of the Berkeley parser and Charniak parser; but reduced the performance of the Stanford parser and Bikel parser. The inconsistency may be due to the small sample size of the ProgressNotes treebank itself. For the DischargeSummaries corpus, direct combination of WSJ and clinical corpora lead to the decrease of the performance for all the four parsers except the Charniak parser. For the MiPACQ corpus, which is 10 times larger than the ProgressNotes corpus, direct combination of WSJ and clinical corpora marginally but consistently improved the performance for all the four parsers (increases of BF ranging from 0.05% -0.43%). These results suggest that it is possible to leverage existing corpora in the open domain to improve parsing of clinical text. However, instead of simply combining different corpora, sophisticated methods, such as domain adaptation techniques, should be investigated to improve parsing in the medical domain. Furthermore, we are also interested in semi-supervised learning methods such as co-training, which may help build large-scale clinical corpus from unlabelled data.

Compared to the default setting, the Stanford deep learning based parser outperformed the Stanford parser after retraining on clinical corpus. In the default setting, the deep learning based parser achieved a better BF than the Stanford parser on the progress notes treebank but a worse BF on the MiPACQ treebank. After retraining on clinical treebanks and generating word embedding vectors from a large clinical corpus, the deep learning based parser achieved better results than the Stanford parser on both clinical treebanks. More specifically, in the ProgressNotes treebank, retraining on the clinical treebank improved deep learning based parsers' BF score by 7.01% on progress notes treebank and 10.72% on MiPACQ treebank. In contrast, for the Stanford parser, the numbers were 3.39% and 9.93%. For both parsers, the increase of the BF on MiPACQ corpus was comparable, however, the deep learning based parser had much more improvement on the progress notes corpus. These findings suggest that using the word embeddings generated from a large-scale clinical corpus has more effects on the progress notes than MiPACQ corpus in this study.

When existing parsers were directly applied to clinical text, a main category of errors was the failure to recognize structures of clinical sentences. We also analysed errors from parsers re-trained on clinical corpora and categorized them into the following major groups:

1) Ambiguity of coordination: For example, in the sentence “Current medications are Keppra 1500 bid and Tegretol - XR 400 bid”, “Keppra 1500 bid” and “Tegretol - XR 400 bid” are both drug with signatures and they should be coordinated. However, in the parsed result, “bid” and “Tegretol” are coordinated.

2) Ambiguity of prepositional phrase (PP) attachment: For example, in the sentence “He denies any problem with chest pain, dyspnea on exertion at this time”, the parser did not identify the prepositional phrase ‘on exertion’ as a modifier to ‘dyspnea’. Clinical knowledge will be useful for solving this type of ambiguity.

3) Errors in the non-terminal symbol ‘NX’: NX was used to mark the head noun within a complicated noun phrase in the annotation guideline. However, parsers had trouble identifying them correctly.

3.5 Conclusion

In this study, we evaluated and compared four state-of-the-art parsers on two types of clinical treebanks. We found that training on the clinical treebank could largely improve the performance of the parsers on clinical text. Among all the parsers, the Berkeley parser achieved the best performance when it was retrained on the MiPACQ corpus.

Chapter 4. Using semantic information to resolve ambiguities of PP attachment and coordination

In the previous chapter, annotated clinical treebanks were used to re-train the state-of-the-art parsers for processing clinical sentences. Although the performance of parsers is improved through re-training on clinical corpora, there are still errors that could be further improved through more advanced methods. Based on the error analysis, two major groups of errors were identified: 1) ambiguity of PP attachment and 2) ambiguity of coordination. In this chapter, I describe our studies on developing new methods for resolving these two types of ambiguity for syntactic parsing of clinical text.

4.1 Using semantic information to resolve PP attachment ambiguity

4.1.1 Introduction

As described in Chapter 1, there are extensive studies on resolving PP attachment errors in the open domain. However, limited work has been conducted in the medical domain. In clinical text, PP attachment usually describes the relation between a clinical event and its attributes. For example, in the sentence "Patient given lab results from 10-03-10", PP "from 10-03-10" should attach to the noun phrase "lab results", which means that the date of lab result is "10-03-10". However, if PP attaches to the verb "given", it will mean the action of "given" happens on "10-03-10". Correctly understanding such relations (e.g. clinical event and its temporal information) is meaningful and highly desired in many clinical studies such as detecting adverse drug events, drug-drug associations etc.

Semantic information has often been used to resolve PP attachment ambiguity in the open domain. There, researchers mainly make use of word meanings to help resolve the ambiguity. However, it sometimes creates new noise as well due to the ambiguous word sense itself. According to the sub-language theory, medical text has more restricted semantic patterns, as compared to general English, making semantic information more effective to help resolve the ambiguity of sentence structures. Nevertheless, the annotation of semantic information is usually not available. Therefore ,automatic solutions to generate high-quality semantic information are also required. Fortunately, over the past decade, more and more clinical NLP systems have been developed and they can provide semantic information needed for resolving PP attachment. In this study, we report our first attempt to leverage semantic information to resolve PP attachment ambiguity in clinical text.

4.1.2 Methods

4.1.2.1 Data sets

In this study, two treebanks were used: 1) the MiPACQ treebank described in Albright et al. [25] and 2) the DischargeSummaries treebank developed locally (see Chapter 2). Sentences with less than 5 tokens from both clinical treebanks were excluded, because they are often section headers and do not require full parsing. After filtering, we retained 10,661 sentences in the MiPACQ treebank and 4,594 sentences in the DischargeSummaries treebank.

There are different types of PP attachment in clinical text. Following previous studies[16, 56, 75], we limited the scope of this study to the most common scenario of PP attachment ambiguity - (V, N1, P, N2), where V stands the verb that precedes the prepositional phrase, P is the prepositional word (e.g. “of”, “in” etc.), N1 is the noun phrase that precedes the prepositional phrase, and N2 refers to the noun phrase in the prepositional phrase. So the ambiguity is that P can either attach to V or to N1. From the above corpora we identified 4,724 sentences in MIPACQ and 2,254 sentences in the DischargeSummaries corpus that contain the (V, N1, P, N2) structure. These sentences were used to develop and evaluate our PP attachment disambiguation methods.

4.1.2.2 Experiments

Because of its superior performance in our previous studies, we decided to use the Berkeley parser in this study. A five-fold cross validation method was used to develop and evaluate our proposed methods. At each round, four-fold of data were used to re-train the Berkeley parser and instances containing (V, N1, P, N2) within the 4-fold of data were used to train our PP attachment classifiers, and the remaining one-fold of data was used for evaluation. Accuracy of the PP attachment classifiers was calculated for each round and then its average reported.

Two models including the Back-Off model used by Collins [56] and a newly developed machine learning based model were developed here and evaluated in this study. Effects of different features used for the classifiers were also evaluated and reported. Once errors of PP attachment were founded in the Berkeley parser, we automatically corrected them by following some rules.

4.1.2.3 PP attachment classification

Backed-Off model

In detecting errors of the PP attachment, we built a baseline system using the backed-off model, which is one of the most classical and popular methods in the English domain. However, we made a slight modification to Collins' work [56], instead of using the headword of noun phrases to determine the attachment, we used the norm form of the headword. We normalized the headword in two different ways: 1) normalizing the headword to its semantic type as determined by our dictionary-based semantic tagger; and 2) if no semantic tag information was related to the headword, normalizing it into the root form using a stemming algorithm [76].

Machine learning based model

In our machine learning based method, given a 4-tuple of the form (V, N1, P, N2), the goal is to classify it as either adverbial attachment (attaching to V) or adjectival attachment (attaching to N1). The features we used include 1) V, P and headword of two noun phrases in each instance; 2) semantic tags of headwords for N1 and N2 based on a lexicon dictionary that was derived from a subset of semantic categories in UMLS (e.g. "DISORDER", "THERPROCDEV" etc.) ; 3) semantic tags of headwords for N1 and N2 based on MedNET system, which identifies three types of named entities including problem, treatment, and test [14]; and 4) the words surrounding N1 and N2 within the window size of 5. Table 7 shows examples of different types of features. The lib-linear SVM classifier was used here [77].

Table 7. An example of features in error detection classifier

Sentence	The patient has had a mild anemia for the last several years .
Headword feature	Prepositional word: for Verb: had N1: anemia N2: years
Dictionary based semantic feature	Tag of N1 headword: DISORDER Tag of N2 headword : TIME
Machine learning based semantic feature	Tag for N1 headword: problem Tag for N2 headword: None
Context words	Context words for N1: [Start] The patient has had + for the last several years Context words for N2: had a mild anemia for + . [end]

After a PP attachment error was detected, we automatically fixed it by removing the original attachment and attaching it to the correct constituents. For example, if the PP attached to the verb in the parsed result, we removed the attachment and attached the PP to the noun phrase and vice versa. Figure 3 shows a parsed result before and after fixing the PP attachment error.

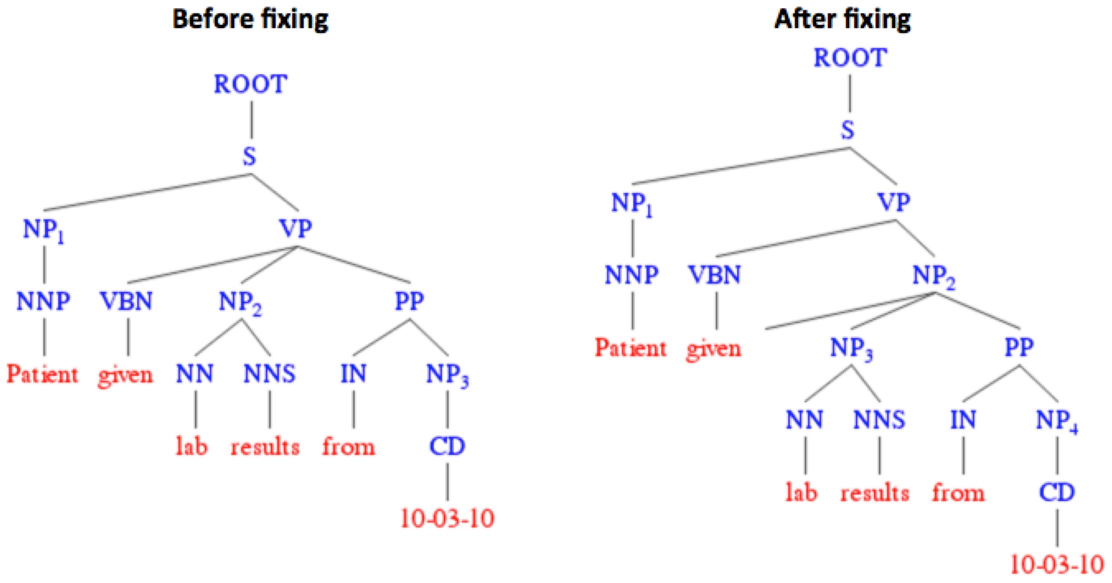


Figure 3. Before and after fixing PP attachment error

4.1.3 Results

Table 8 shows the accuracy of PP attachment for the original parsed results generated by the Berkeley parser, back-off model and different feature sets for the machine learning based model on the MiPACQ treebank and the DischargeSummaries treebank. The machine learning based model achieved better results than the original parser and the back-off model. For the MiPACQ treebank, the best precision was achieved by the machine learning based method with all features were used.

Table 8. Accuracy of PP attachment

Experiment	MiPACQ	DischargeSummaries
Berkeley parser	0.7932	0.7607
Backed-off model	0.7939	0.7586
Head feature	0.7994	0.7617
Head + dictionary based semantic	0.8082	0.7693
Head + dictionary based semantic + machine learning based semantic	0.8105	0.7745
Head + dictionary based semantic + machine learning based semantic + context words	0.8167	0.7784

4.1.4 Discussion

We conducted the first study to resolve the PP ambiguity when parsing clinical text. The results were promising. Our machine-learning based model that utilizes semantic features improved the Berkley parser's performance on handling the (V, N1, P, N2) structures by 2.35% on the MiPACQ treebank and 1.77% on the DischargeSummaries treebank respectively.

Semantic information helps to resolve PP attachment ambiguity. The dictionary based semantic information increased the accuracy by 0.88% and 0.76% on two treebanks respectively. Moreover, adding the feature of semantic information generated from MedNET further increased the performance by 0.23% and 0.52%, probably due to the difference between semantic information extracted by terminologies and NLP systems. We also noticed that MedNET helps more on the DischargeSummaries treebank compared with the MiPACQ Treebank, probably because MedNET was trained on the i2b2 corpus.

We also conducted additional experiments to assess the effect of resolving (V, N1, P, N2) ambiguity on the overall parsing performance. Our results show that there is an improvement, but very limited – an increase of 0.03% on the MiPACQ treebank and 0.07% on the DischargeSummaries treebank. We noticed several reasons behind this finding. First of all, we only dealt with (V, N1, P, N2) ambiguity and ignored other types of PP attachment ambiguity. According to our analysis, about 44.3% MiPACQ sentences (4,724 instances in 10,661 sentences) and 49.0% i2b2 sentences (2,254 instances in 4594 sentences) contain PP attachments. We can potentially develop methods to improve other types of PP attachment ambiguity. In addition, the current measurement of parsing (F-1 score) is highly related to the length of the sentences. As the sentences containing (V, N1, P, N2) structure tend to be long sentence, the change of F-1 score on these sentences is relatively low.

4.1.5 Conclusion

In this section, by leveraging various types of semantic information, we developed a machine learning based solution to increase the accuracy of identifying PP attachment with the form of "V, N1, P, N2" in the parsed results and further improved the parsing result.

4.2 Using semantic information to resolve coordination ambiguity

4.2.1 Introduction

Coordination ambiguity is another common issue when parsing clinical text. For example, the sentence "Current medications are Keppra 1500 bid and Tegretol - XR 400 bid" describes a list of medication that includes names of drugs and their signature information, which is very often seen in the clinical text. However, such coordination structures are sometimes not easily recognized correctly by the syntactic parser. Figure 4 shows one possible wrong parse tree of the above sentence. Another more complicated example of coordination ambiguity is the sentence “Past surgical history includes remote tonsillectomy; hemorrhoidectomy; appendectomy; a partial gastrectomy, vagotomy, and cholecystectomy in1973; bilateral inguinal hernia repair; penile prosthesis in 1984 ; and open reduction and internal fixation of right hip fracture in 1989.” , which contains a long list of clinical procedures and the syntactic parser has difficulty identifying the coordination structures .

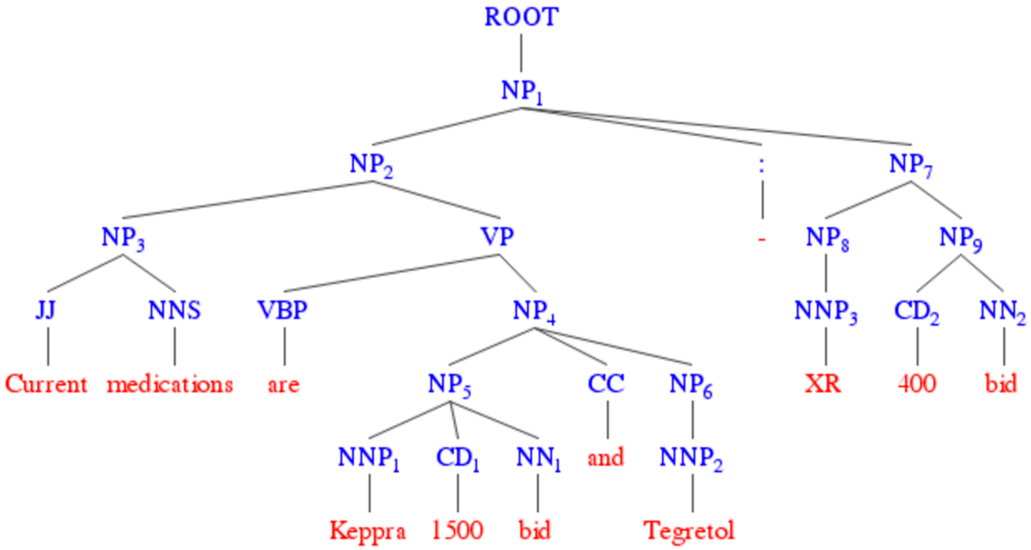


Figure 4. An example of a wrong parse tree of a sentence with coordination ambiguity.

Semantic information, integrating abundant domain knowledge, can play an important role in resolving coordination ambiguities in the clinical corpus. For the above two examples, the main reason why a syntactic parser cannot correctly understand the coordination structure is due to lack of clinical knowledge. In the first example, if semantic information is provided as shown in Figure 5, such a coordination error can be avoided by the parser. Similarly, semantic information could also help identify the correct coordination structure in the second example.

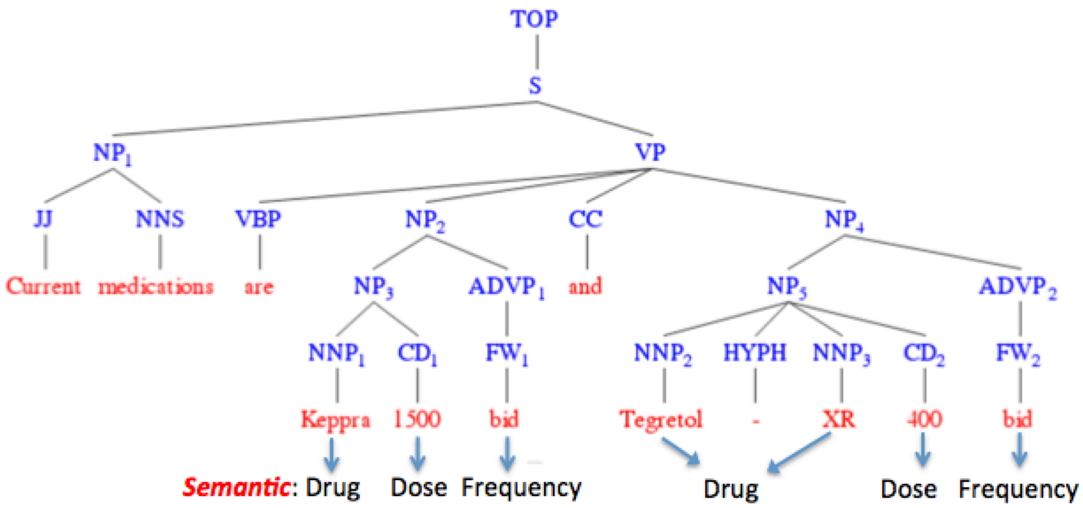


Figure 5. An example of how semantic information could help resolve coordination ambiguity in clinical text.

To the best of our knowledge, no study has specifically focussed on resolving coordination ambiguity in clinical text. In this section, we describe a new method that we have developed to resolve coordination ambiguity using semantic information.

4.2.2 Methods

4.2.2.1 Data

The same two treebanks: MiPACQ and DischargeSummaries (see more details described in section 4.2.1.1.) were used in this study. For the MiPACQ treebank, we split it into a development dataset and a test dataset. The development corpus contained 5,000 sentences and the test corpus contained 5,661 sentences. We reviewed the development corpus to generate rules for coordination error detection and fixing and then applied these rules to the test corpus and reported the evaluation results. Moreover, to test the generalizability of the rules, we also applied the rule-based coordination disambiguation system to the DischargeSummaries treebank and reported its performance.

All coordination structures from two treebanks were identified by searching the part-of-speech tag of “CC” in each sentence, which results in a collection of 4,529 sentences and 2,116 sentences from the MIPACQ and the DischargeSummaries treebanks, respectively. As an initial attempt, we limited this study to a common type of coordination (N1 CC N2), where N1 denotes a noun phrase, CC denotes the conjunction word, and N2 refers to another noun phrase. Based on these criteria, we identified 2,545 sentences in the MiPACQ and 1,021 sentences in the DischargeSummaries respectively, for the proposed study.

4.2.2.2 Experiments

We used the Berkeley parser again to generate baseline parse trees for clinical sentences used in this study. The coordination disambiguation method that we developed is a rule-based system that consists of two steps: 1) detecting potential errors of coordination structures generated by the Berkeley parser; and 2) fixing detected errors by searching top candidate alternative parse trees. Semantic information used in the specified rules was from a local clinical NLP system called CLAMP (http://clamp.uth.edu/), which identifies not only medical entities (e.g. “problem”, “treatment”, “test” etc.) but also their modifiers/attributes (e.g. “body location”, “drug form”, “lab test value” etc.).[14, 78]

To evaluate the coordination error detection method, we manually reviewed errors that were detected by our system and reported accuracy. For the error fixing step, we evaluated the parse trees on identified sentences and reported *F*1-scores for both before and after fixing coordination errors.

4.2.2.3 Coordination disambiguation system

The first step was to identify potential errors in coordination following the (N1 CC N2) structure. Our assumption was simple: if the headwords of N1 and N2 share the same semantic type, N1 and N2 should be coordinated into a single element. Based on the outputs of CLAMP, we first checked whether N1 and N2 are same type of medical entities or attributes. If so, then we further checked if they are coordinated into one element in the parse tree – we f identified the lowest common ancestor of two noun phrases in the parsing tree and made sure that the path between the lowest common ancestor and the noun phrase contained NP only. For example, in Figure 6, both “colitis “ and “other bowel pathology” are tagged as “problem”, the lowest common ancestor of them is NP_3, there is no node on the path between NP_3 and NP_4 and the node on the path between NP_3 and NP_6 is a noun phrase (NP_5). Therefore these two noun phrases were correctly coordinated in the example.

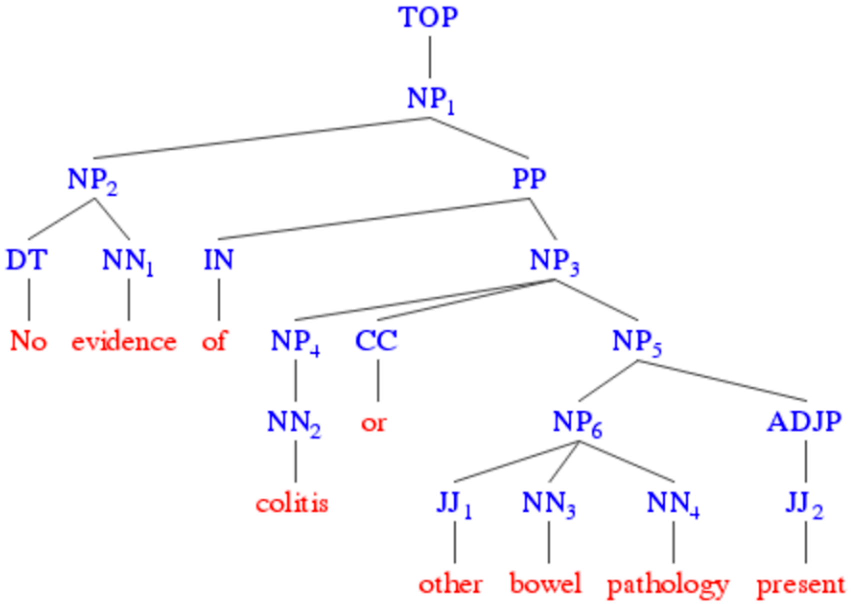


Figure 6. Example of demonstration on the rules to identify errors in coordination

Fixing detected errors of coordination is not straightforward, as it involves re-generating the parse tree. We proposed a workaround by using a re-ranking-like approach. The traditional machine learning based re-ranking method is a process of using global features of a parsing tree to re-rank the top n-best parse trees to identify the most probable one. In our approach, we just searched the top 50-best parse trees and selected an alternative parse three that had the highest score but did not have the identified coordination error. If the coordination error occurred in all candidate parse trees, we kept the original parse tree.

4.2.3 Results

Table 9 shows the results of error detection. For the development corpus in the MiPACQ treebank, 34 errors of coordination attachment were detected and the accuracy was 100% according to our manual review. There were 34 errors found in the test corpus in the MiPACQ treebank and the accuracy was 94.9%. For the DischargeSummaries treebank, there were 62 errors detected and the accuracy was 90.4%.

Table 10 shows the results of error fixing. For the development corpus in MiPACQ treebank, among 34 errors detected, 28 were fixed and improved the F score of the identified sentences of parsing from 68.66% to 79.01%. For the test corpus in MiPACQ Treebank, 34 out of 39 errors were fixed and the *F*1-score was increased from 74.35% to 80.35%. For the DischargeSummaries treebank, 56 errors were fixed and the *F*1-score was increased from 70.96% to 74.54%.

Table 9. Result of error detection on coordination

	MiPACQ development	MiPACQ test	DischargeSummaries
# of errors detected	34	39	62
Accuracy	100%	94.9%	90.3%

Table 10. Result of error fixing on coordination

	MiPACQ development	MiPACQ test	DischargeSummaries
# of errors detected	28	34	56
Original $F1$-score of identified sentences	68.66%	74.35%	70.96%
Improved $F1$-score after fixing the error	79.01%	80.35%	74.54%

4.2.4 Discussion

In this study, we conducted a study to resolve the coordination ambiguity when parsing clinical text. We developed a rule-based algorithm to detect one types of coordination errors and leveraged top n-best parse trees to fix identified errors. Our evaluation showed that the proposed approach can improve the parser's performance of handling sentences with the specific type of coordination errors, even when applied to a different corpus (the DischargeSummaries dataset).

In addition, we also evaluated the effect of this approach on the overall parsing performance. We found that the Berkley parser's performance was increased from 85.97% to 86.03% on the MiPACQ test corpus and from 84.45% to 84.51% on the entire DischargeSummaries corpus, after integrating our coordination disambiguation method. Although the improvement is small, which is expected as we focussed on one type of coordination errors only, it demonstrated its potential to further improve syntactic parsing of clinical text.

The study conducted here was just a start to demonstrate the potential of coordination disambiguation. To provide further insights into the potential challenges, we conducted an additional analysis to group the coordination errors by 5 semantic types including disease, medication, procedure, lab test and body location. As Table 11 shows, for the MiPACQ test corpus, most of the detected errors were about clinical diseases, accounting for 61.5%. For the DischargeSummaries corpus, most of them were also about the clinical diseases, accounting for 51.6%.

Table 11. Statistics on coordination errors by semantic types

	All	**Disease**	**Medication**	**Procedure**	**Lab test**	**Body location**
MiPACQ test	39	24 (61.5%)	5 (12.8%)	4 (10.2%)	3 (7.7%)	3 (7.7%)
DischargeSummaries	62	32 (51.6%)	4 (6.45%)	6 (9.68%)	8 (12.9%)	12 (19.35%)

We also conducted error analysis and found that the false positives of the error detection had several causes. One was related to wrong semantic tags: for example, as Figure 7 shows, in the sentence of "he was started on Lasix for diuresis and his Captopril was increased for greater afterload reduction", "diuresis" was wrongly identified as a treatment and "diuresis and his Captopril" was thus identified as a coordination structure. We also noticed coordinations at the clause level, which is not handled by the current approach. Figure 8 shows such an example, where "single phototherapy" is the object of the previous clause and "phototherapy" is the subject of the latter clause, in the sentence of "Infant decreased to single phototherapy and phototherapy was discontinued on day of life six .".

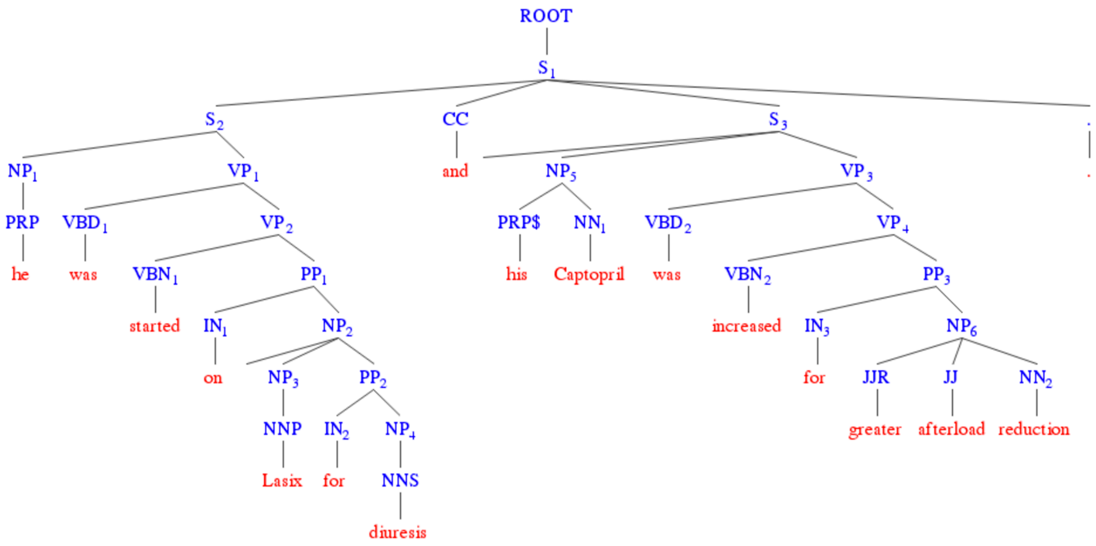


Figure 7. Example of error semantic type

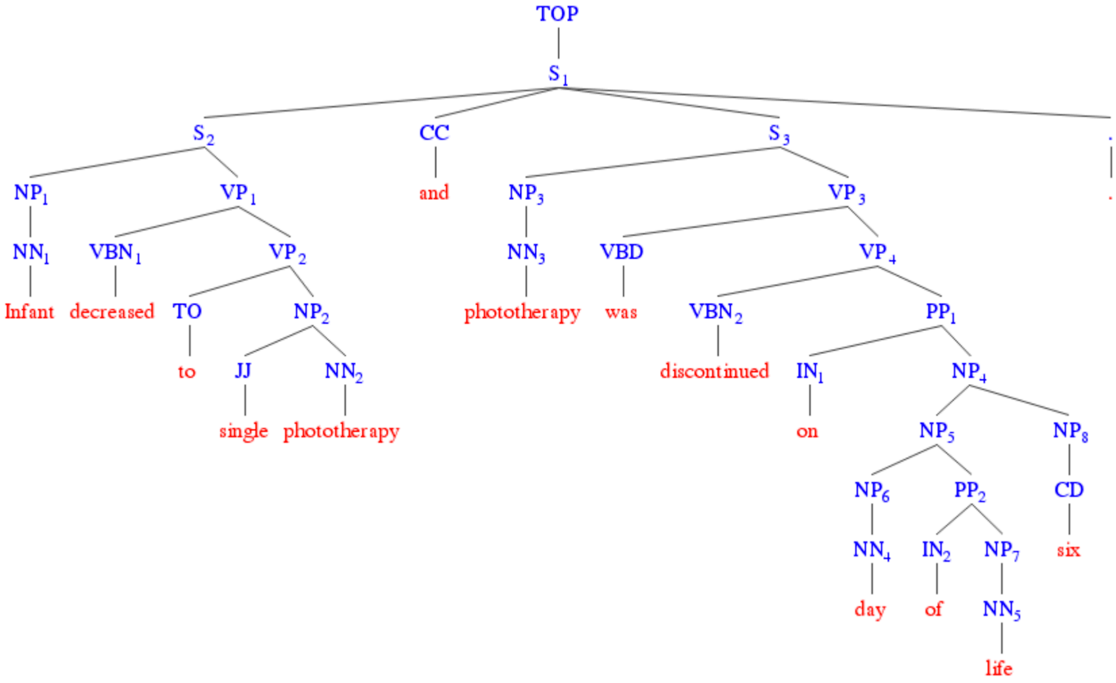


Figure 8. Example of sentence coordination

4.2.5 Conclusion

In this section, we developed a rule-based solution to identify one certain type of coordination error and we leveraged top n-best parse trees to fix the coordination error and further improved the parsing performance.

Chapter 5. Uses of improved syntactic parsers on other NLP tasks

In the previous chapters, we demonstrated improved performance of syntactic parsers on clinical text. To further demonstrate the use of such improved syntactic parsers for other clinical NLP tasks, we describe our studies about applying our parsers to two additional NLP applications: 1) semantic role labeling; and 2) temporal relation extraction.

5.1 Apply syntactic parsers on semantic role labeling task

5.1.1 Introduction

In the biomedical domain, semantic relation extraction systems, such as LSP [46], MedLEE [1], MedEx [3] for clinical text and SemRep [79, 80] for biomedical literature, have shown good performance and been widely used in different applications. These early-stage systems were often based on manually extracted patterns, following the sub-language theory [47]. Based on the sub-language theory, the language of a closed domain (e.g., medicine and biomedicine) has special syntactic patterns as well as a limited number of main semantic types. Therefore, possible semantic relations could be identified by restricted constraints of syntactic and/or semantic patterns [8]. However, a careful examination of syntactic alterations that express the same semantic relations in biomedical text revealed that even in a semantically restricted domain, syntactic variations are common and diverse [81]. Thus, the coverage and scalability of manually extracted patterns may not be sufficient for those syntactic variations. In recent years, promoted by increasing challenges held by different portals (e.g., BioCreative, BioNLP, I2b2 and SemEval), more and more automatic information extraction systems have been built for different biomedical subdomains using data-driven statistical methods, such as machine learning algorithms. However, diverse syntactic variations still remain as an essential problem to extract semantic information from biomedical text, especially for clinical text, which contains more fragments and ill-formed grammars.

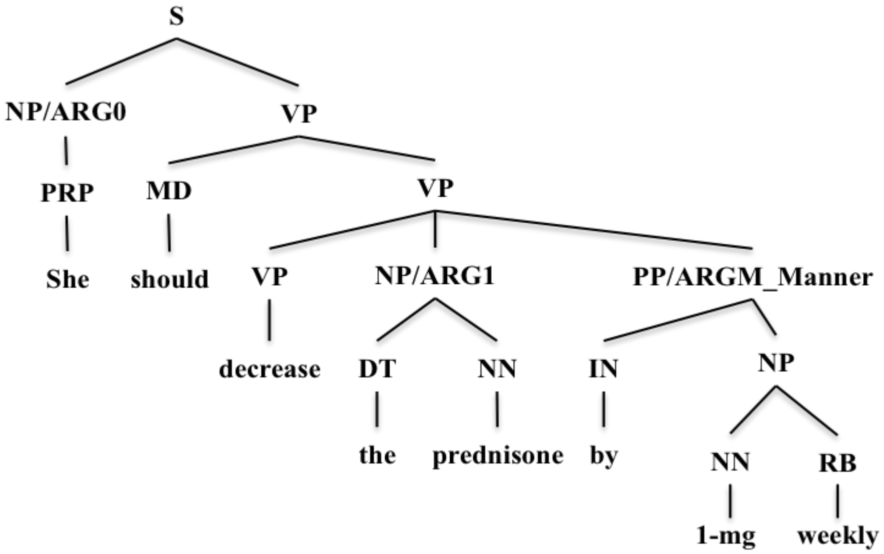


Figure 9. A syntactic parse tree with semantic roles added (ARGs)

One potential solution to this problem is semantic role labeling [82] (SRL) (also known as shallow semantic parsing), which focuses on unifying variations in the surface syntactic forms of semantic relations. Specifically, the task of SRL is to label shallow semantic relations in a sentence as predicate argument structures (PAS) [83]. A predicate usually refers to a word indicating an event or a relation, and arguments (ARGs) refer to syntactic constituents representing different semantic roles in the event or relation. For each predicate, arguments representing the most important semantic roles are labeled with numbers, usually from ARG0 to ARG5. In addition, arguments representing modifiers of events (i.e., location, time, manner, etc.) are labeled as ARGMs. Taking the sentence "She should decrease the prednisone by 1-mg weekly" in Figure 9 as an example, the verb phrase "decrease" is the predicate indicating the event; the noun phrase

"She" represents the role of ARG0, indicating the initiator/executor of the action "decrease"; the noun phrase "the prednisone" represents the role of ARG1, indicating the receptor of the action “decrease” (i.e. the entity decreased); while the prepositional phrase "by 1-mg weekly" represents the manner of how to decrease the prednisone (ARGM-Manner).

Shallow semantic relations, or PASs are usually applied as features for machine learning algorithms, sentence structural representations in kernel-based models or inference rules in different applications, including question answering, text summarization and information extraction [84-87], etc. Specifically, PASs have been investigated in various biomedical sub-domains [88-90] and made positive contributions in semantic information extractions, such as extracting drug-drug interactions from biomedical literature[90] and temporal relations from clinical text [91].

Generally, a typical SRL system is built by using machine-learning methods based on annotated corpora. Since semantic roles are formed by syntactic constituents, two corpora are needed to build SRL systems, namely a corpus of syntactic parse trees and a corresponding corpus of semantic roles annotated on it. The most widely used large-scale corpora in the open domains are the Penn Treebank [92] and the SRL corpus PropBank[83] developed on it. Many state-of-the-art syntactic parsers [35, 38, 93] have been developed and applied to SRL in the open domains [94-96]. Some previous studies attempted to adapt these parsers (e.g., the Stanford Parser) to clinical text using medical lexicons [54, 97]. Recent years have also seen emerging efforts for syntactic annotation guidelines and corpora of clinical text [98]. For example, the MiPACQ corpus (a multi-source integrated platform for answering clinical questions) annotated syntactic trees for 13,091 sentences following the Penn Treebank Style. Furthermore, several SRL corpora were developed for clinical text following the PropBank Style. The available corpora were of different genres and note styles, including operative notes [99], radiology notes [100] from the SHARP Area 4 project (Strategic Health IT Advanced Research Projects), colon cancer pathology and clinical notes from the MiPACQ corpus[53] and the THYME corpus [100] (Temporal Histories of Your Medical Events). Based on those corpora, studies have been conducted to investigate SRL techniques for clinical text from EHRs. Albright et al. (2013) and Zhang et al. (2014) developed SRL systems on the MiPACQ corpus using dependency parse trees and constituent parse trees, respectively. Wang et al. (2014) built a SRL system on operative notes using an adapted parser.

Given that semantic roles are formed by syntactic constituents in the sentence, an effective parser to first recognize those syntactic constituents is critical for developing a practical SRL system [101]. Furthermore, an effective feature set to describe the syntactic patterns between the predicate and the argument is also essential to SRL. Although previous work has compared different syntactic parsers and representations for biomedical event extraction from literature [101], there are no formal evaluations and comparisons of state-of-the-art parsers [38, 99, 102], and features [53, 102], for SRL in the medical domain.

In this study, we evaluated the SRL performance of three state-of-the-art constituent syntactic parsers: the Stanford parser, the Charniak parser [93] and the Berkley parser [39], using the MiPACQ corpus. We focused on constituent parse trees here because they could be directly converted to dependency parse trees [103]. The purpose of this study was two-fold: (1) to evaluate the SRL performance of existing state-of-the-art English parsers on clinical text, both the original parsers developed on Penn Treebank and parsers retrained on the clinical Treebank; and (2) to validate the effectiveness of state-of-the-art syntactic features for SRL in the open domain [104] and the biomedical domain [105, 106] on clinical text. To the best of our knowledge, this is the first comprehensive study that investigated the influence of syntactic parsing and features for SRL on clinical text using multiple state-of-the-art parsers.

5.1.2 Methods

5.1.2.1 Dataset

This study used the MiPACQ dataset for SRL experiment. MiPACQ is built from randomly selected clinical notes and pathology notes of Mayo Clinic related to colon cancer [53]. Layered linguistic information is annotated in MiPACQ, including part of speech (POS) tags, syntactic Treebank, PASs for SRL, named entities, and semantic information from Unified Medical Language System. The syntactic Treebank annotations in MiPACQ follow the Penn Treebank guidelines, and the predicate-argument structure annotations for SRL follow PropBank guidelines. 13,091 sentences are annotated with syntactic trees. Among them, 6,145 sentences in MiPACQ are annotated for SRL, including 722 verb predicates with 9,780 PASs and 415 nominal predicates with 2,795 PASs.

5.1.2.2 The basic SRL system

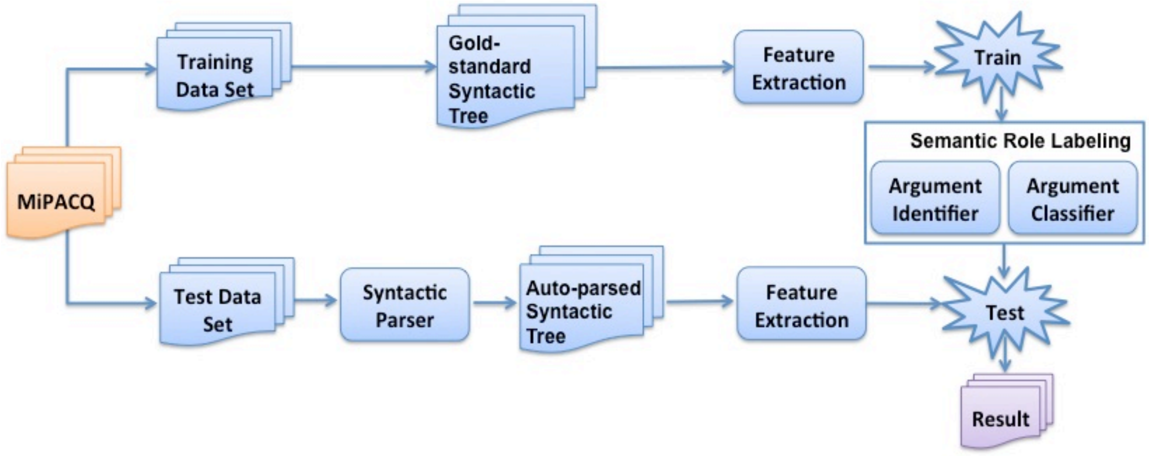


Figure 10. Study design for semantic role labeling of clinical text

Figure 10 shows the study design for SRL of clinical text. Basically, the SRL system can be partitioned into the training stage and the testing stage. In the training stage, gold-standard syntactic trees of the training data set annotated in MiPACQ are used for feature extraction. A SRL task consists of two sub-tasks, the argument identification sub-task and the argument classification sub-task. First, a binary non-Argument vs. Argument classifier is built as the argument identifier on the entire dataset for all the predicates, instead of building one model per predicate. For argument classification, a multi-class classifier is built to assign semantic roles to arguments of all the predicates. In the testing stage, syntactic trees automatically generated by the syntactic parser are used for feature extraction. For each predicate, the argument candidates first go through the argument identifier. If one candidate is identified as an argument, it will go through the argument classifier that assigns the semantic role.

5.1.2.3 Comparing syntactic parsers and features

Three widely used state-of-the-art syntactic parsers, the Stanford parser [38], the Charniak parser[93], and the Berkley parser[39] were investigated for their influence on the SRL performance in our study. Moreover, state-of-the-art features, most of which are syntactic features, commonly used in the open domain and biomedical domain were extracted and compared for use with clinical text.

Features

Similar to previous work of SRL for biomedical literature and clinical text[102], we adopted the common features used in current state-of-the-art SRL systems. The features include baseline features from the original work of Gildea and Jurafsky (2002)[107], advanced features taken from Pradhan et al. (2005)[82] and feature combinations from Xue and Palmer (2004)[94].

The features can be categorized into three major groups: (1) basic features include the lexical and syntactic features of the predicate and the argument; (2) context features include features of the surrounding syntactic nodes and the syntactic paths between the predicate and the argument; (3) feature combinations are feature tuples formed of two unitary features from the previous two groups. The complete feature set is described in Table 12. Except for the lemmatization of the predicate word and the relative position between the argument and the predicate, all the rest of the features are at the syntactic level and need to be extracted from the parse tree. For clarity, Table 13 lists the specific features extracted for the argument candidate “1-mg weekly” of the predicate “decrease ” in the example sentence shown in Figure 9.

Table 12. Feature list of semantic role labeling

Feature Group	**Description**
basic features	
Predicate	Lemmatization of the predicate word Voice of the verb predicate, i. e., active or passive
Argument	Syntactic head, first word, last word of the argument phrase and their POS tags Syntactic category of the argument node Whether the argument is a preposition phrase Enriched POS of prepositional argument nodes (e. g., PP-for, PP-in)
Relative position	Relative position of the argument with respect to the predicate (before or after)
Context features	
Production rule of predicate	Production rule expanding the predicate parent node
Syntactic category of argument neighbors	Syntactic categories of the parent, left sister and right sister of the argument node
Path	Syntactic path linking the predicate and an argument
No-direction path	Like Path, but without traversal directions
Partial path	Path from the argument to the lowest common ancestor of the predicate and the argument
Syntactic frame	Position of the NPs surrounding the predicate
Feature combinations	Predicate and headword of the argument Predicate and Syntactic category of the argument Predicate and relative position Predicate and path

Experiments

PASs with at least one argument were used for the experiment. We used the open source toolkit Liblinear [77] as implementations of the support vector machine algorithm. For each implemented method, all parameters were tuned for optimal performance.

Experiments and systematic analysis were conducted as follows:

1) Evaluate SRL performance of parsers with their default settings: In this experiment, we directly applied the three parsers to process all sentences of the test dataset. All the

parsers were invoked with their default settings and models, which had been trained on the Penn Treebank.

Table 13. An example of features extracted for semantic role labeling

Sentence: She should decrease the prednisone by 1-mg weekly Predicate: decrease Argument candidate: 1-mg weekly	
Feature Group	Feature value
Basic features	
Predicate	decrease active
Argument	hw_1-mg, hw_pos_NN, fw_by, fw_pos_IN, lw_weekly, lw_pos_RB PP Yes PP-by
Relative position	after
Context features	
Subcategory of predicate	VP→VB–NP–PP
Syntactic category of argument neighbors	scp_VP, scl_NP, scr_null
Path	PP↑VP↓VB
No-direction path	PP_VP_VB
Partial path	PP↑VP
Syntactic frame	Position of the NPs surrounding the predicate
Feature combinations	decrease_1-mg decrease_PP decrease_after decrease_VB↑VP↓PP

2) Evaluate SRL performance of parsers re-trained on the clinical Treebank: To assess if the annotation of clinical Treebank could improve the performance of SRL, we applied three parsers retrained on the MiPACQ Treebank. We conducted ten-fold cross validation evaluation for each parser. The cross-validation involved dividing the clinical corpus equally into 10 parts, and training the parser on 9 parts with testing on the remaining part

each time. We repeated the same procedure 10 times, one for each part, and then combined the results from the 10 parts to report the performance.

3) Evaluate SRL performance of each syntactic feature: To validate if syntactic features commonly used in the open domain were effective for clinical text, we conducted multiple runs of experiments, adding one new syntactic feature into the feature set for each run. The experimental results were compared to check the effectiveness of each feature.

Evaluation

Precision (P), recall (R) and *F*1-measure (*F*1) were used as evaluation metrics for argument identification (AI) and combined SRL task. Precision measures the percentage of correct predictions of positive labels made by a classifier. Recall measures the percentage of positive labels in the gold standard that were correctly predicted by the classifier. *F*1-measure is the harmonic mean of precision and recall. During the process of argument classification (AC), the boundaries of candidate arguments are already identified by the argument identification step. Therefore, the accuracy (Acc) of the classifier was used for evaluation, which is defined as the percentage of correct predictions with reference to the total number of candidate arguments correctly recognized in the argument identification step. Ten-fold cross validation was employed for performance evaluation.

5.1.3 Results

Table 14 illustrates the performance of semantic role labeling systems, which were trained on the gold standard syntactic trees and tested on the parsed results of Stanford, Charniak and Berkley, as well as the gold standard syntactic trees, respectively. For these experiments, the whole feature set described in Table 13 was used. The original parsers trained on the Penn Treebank produced relatively lower performance. Charniak got the lowest *F*1-measure of 61.40%, whereas Berkley outperformed the other two parsers with a *F*1-measure of 68.15%. After retraining on the clinical Treebank, the performance of all three parsers increased significantly, with the optimal *F*1-measure of 71.38% achieved by Berkley. Testing on the gold standard parse trees yielded a *F*1-measure of 82.13%.

Table 14. Performance of semantic role labeling systems trained on the gold standard syntactic trees and tested on the parsed results of Stanford, Charniak and Berkley, and the gold standard syntactic trees, respectively (%)

Parser	Model	AI			AC	AI+AC		
		P	*R*	F_1	*Acc*	*P*	*R*	F_1
Stanford	Default	70.4	82.1	75.8	88.1	62.0	72.4	66.8
	Retrained	75.7	85.2	80.2	88.0	66.6	75.0	70.6
Charniak	Default	67.7	74.5	70.9	86.5	58.6	64.4	61.4
	Retrained	74.2	87.0	80.1	87.9	65.2	76.6	70.4
Berkley	Default	72.8	83.0	77.6	87.7	63.9	72.9	68.1
	Retrained	76.7	85.3	80.8	88.3	67.7	75.4	71.3
Gold Standard		91.4	91.6	91.5	89.7	82.0	82.2	82.1

To investigate whether syntactic features commonly used in the open domain are also effective for clinical text, multiple experiments were conducted by adding one new syntactic feature incrementally for each run. Table 15 lists the SRL performance of both the gold standard corpus and the parse results of the retrained Berkley. As the baseline, the first run adopted all the basic features of predicate, argument and their relative position. Numbers in parenthesis show the changes to *F*1-measure of argument identification and accuracy of argument classification by adding each new feature. As illustrated in Table 15, all the syntactic features effective in the open domain were also helpful for argument identification of clinical text. The *F*1-measure was improved consistently from 20.07% and 17.47% to 89.75% and 88.33% for the gold standard corpus and the retrained Berkley parser, respectively. In addition to the basic features, phrase types of argument neighbours, and the three path features made the most contribution to argument identification. The In contrast, for argument classification, the basic features already yielded an accuracy of 86.74% for the gold standard corpus and an accuracy of 83.78% for the retrained Berkley. Since the path features between the predicate and an argument dropped the accuracy slightly, we conducted additional experiments by removing those features for argument classification, which improved the overall $F1$-measure of our SRL systems to 82.14% (vs. 82.13%) for the gold standard corpus and to 71.41% (vs. 71.38%) for the retrained Berkley parser.

5.1.4 Discussion

Effective syntactic parsers and features are critical to establishing a practical SRL system. This study undertook a formal evaluation and comparison of SRL performance on a clinical text corpus MiPACQ, using four state-of-the-art syntactic parsers and common syntactic features used in the open domain. Experimental results demonstrated that retraining parsers on clinical corpora could improve the SRL performance significantly, with an optimal $F1$-measure of 71.41% achieved by the Berkley parser. Despite the telegraphic type of clinical text, state-of-the-art syntactic features in the open domain also proved to be effective for clinical text.

Table 15. Semantic role labeling performance of testing on the gold standard syntactic trees and parsed results of retrained Berkley by adding one new feature each time (%)

Feature Group	Test Corpus	AI			AC	AI+AC		
		P	*R*	F_1	*Acc*	*P*	*R*	F_1
Baseline - Predicate+Argument+ Relative position	Gold	60.57	12.04	20.07	86.74	54.83	10.94	18.23
	Auto-parsed	57.56	10.30	17.47	83.78	52.97	9.48	16.07
Production rule of predicate	Gold	59.73	13.99	22.67 (+2.60)	87.20 (+0.46)	54.54	12.77	20.69 (+2.46)
	Auto-parsed	58.34	10.91	18.38 (+0.91)	84.54 (+0.76)	53.49	10.00	16.85 (+0.78)
Phrase type of argument neighbors	Gold	58.62	23.48	33.52 (+10.85	87.70 (+0.50)	52.87	21.17	30.22 (+9.53)
	Auto-parsed	51.20	20.73	29.49 (+11.11	85.58 (+1.04)	46.11	18.67	26.56 (+9.71)
Path	Gold	88.93	55.97	68.70 (+35.18	87.66 (-0.04)	80.45	50.63	62.14 (+31.92
	Auto-parsed	75.24	57.10	64.91 (+35.42	85.69 (+0.11)	66.20	50.24	57.11 (+30.55
No-direction path	Gold	88.58	60.62	71.97 (+3.27)	87.58 (-0.08)	79.66	54.51	64.72 (+2.58)
	Auto-parsed	74.38	61.09	67.07 (+2.16)	85.44 (-0.25)	64.97	53.35	58.58 (+1.47)
Partial path	Gold	91.13	91.19	91.16 (+19.19	87.60 (+0.02)	79.85	79.90	79.87 (+15.15
	Auto-parsed	76.87	84.92	80.69 (+13.62	85.41 (-0.03)	65.60	72.47	68.86 (+10.28
Syntactic frame	Gold	91.14	91.31	91.22 (+0.06)	87.61 (+0.01)	79.81	79.95	79.88 (+0.01)
	Auto-parsed	76.58	85.52	80.80 (+0.11)	85.43 (+0.02)	65.42	73.07	69.03 (+0.17)
Feature combinations	Gold	91.41	91.60	91.51 (+0.29)	89.75 (+2.14)	82.04	82.21	82.13 (+2.26)
	Auto-parsed	76.72	85.37	80.81 (+0.01)	88.33 (+2.92)	67.77	75.40	71.38 (+2.35)

In terms of SRL errors caused by syntactic parsers, a major category was that the parsers did not recognize a large number of syntactic constituents acting as arguments (Original

Stanford: 1,175, Charniak: 1,377, Berkley: 1,262). Nevertheless, retraining parsers on the clinical Treebank reduced such errors greatly (Retrained Stanford: 887, Charniak: 973, Berkley: 816). Another major type of syntactic problems that caused SRL errors was the essential syntactic structure ambiguities. For example, the sentence "He continues to note the sensation of bilateral leg numbness and pins and needle sensation with walking" contains conjunctive structures linking two phrases "the sensation of bilateral leg numbness", and "pins and needle sensation". It's hard to determine if the prepositional phrase "with walking" only modifies the "pins and needle sensation" or both phrases.

Despite the unique characteristics of clinical text, such as fragments and ill-formed grammars, all the state-of-the-art syntactic features in the open domain contributed positively to clinical text, except for path features that dropped the accuracy of argument classification slightly. One possible reason for this decreased performance is that the specific semantic role of an argument in clinical text is dependent not only on syntactic paths but also on the clinical lexicon and relations. As an example, in the phrase "an advanced breast cancer treated with radiation therapy", "an advanced breast cancer" is annotated as ARG2 (illness or injury) in the gold standard. However, it was mistakenly labeled as ARG1, because the extracted syntactic path features were similar to those of ARG1 in the corpus.

5.1.5 Conclusion

In this section, we made a formal evaluation and comparison of SRL performance on a clinical text corpus, MiPACQ, using three state-of-the-art parsers, the Stanford parser, the Berkley parser, and the Charniak parser and state-of-the-art syntactic features from the open domain. Experimental results validated the effectiveness of retraining parsers with a clinical Treebank. The results also demonstrated that common syntactic features in open domain contribute positively to parser performance on the clinical text.

5.2 Apply syntactic parsers to the temporal relation extraction task

5.2.1 Introduction

Temporal information extraction (TIE) is a challenging area in NLP research; but it is important for many NLP tasks, such as question answering, document summarization, and discourse analysis [108, 109]. For most clinical NLP systems, accurate recognition of the timing of medical events is important for many medical reasoning tasks. A clinical TIE system must be able to identify events, temporal expressions and temporal relations between them to create a complete timeline of medical events for a patient. Although much effort has been made to the representation, annotation, and extraction of temporal information in the general English domain (e.g., the TimeML framework [110]), the state-of-the-art TIE systems still don't perform very well (F-measures around 60–70%) [111, 112]. Moreover, the telegraph style of clinical text made extracting temporal information from clinical text even more difficult than from general English texts.

In the general English domain, many TIE studies are based on natural-language text corpora, such as newswires. TIE work started with temporal representation in the 1980s. A milestone was interval-based algebra for representing temporal information in natural language, proposed by Allen in 1983 [113]. Many early studies adopted Allen's representation, which promptly became a standard. In the 1990s, the widespread development of large annotated text corpora for NLP advanced TIE research rapidly. Community-wide information extraction tasks started to include TIE tasks. The message understanding conferences (MUCs) sponsored by the US government organized two consecutive temporal-related tasks: MUC-6 (1995) [114] and MUC-7 (1998) [115]. In MUC-6, extracting absolute time information (i.e., extracting exactly-specified times in the text) was a part of a general named entity recognition (NER) task. In MUC-7, the TIE task was expanded to include extraction of relative times. These two tasks defined the Timex tags, which interpret time expressions into a normalized ISO standard form through the TIDES Timex2 guidelines [116, 117]. In 2004, extracting and normalizing temporal expressions according to the Timex2 guidelines for both English and Chinese texts was part of the Time Expression Recognition and Normalization Evaluation challenge, sponsored by the Automatic Content Extraction program [118]. These tasks provided preliminary but valuable contributions to TIE research.

In 2004, rapid development of TIE methods started with the work of TimeML [110], a robust specification language for events and temporal expressions in natural language. The TimeML schema mainly integrates two annotation schemes: TIDES (Translingual Information Detection, Extraction, and Summarization) TIMEX2 and STAG (Sheffield Temporal Annotation Guidelines) [119, 120]. It defined three elements of temporal information: events, temporal expressions, and temporal relations. Events, including verbs, adjectives, and nominals, corresponding to events and states are classified into different types, and have various attributes, including tense, aspect, and other features. Temporal expressions are token sequences that denote times with various attributes such as their normalized values. TimeML also represents temporal relations between events/times using an Allen-like format. It defines temporal relations using three types of links: TLinks (Temporal Links), SLinks (Subordination Links), and ALinks (Aspectual Links). TimeML has become an ISO standard for temporal annotation. Several TimeML-based annotated corpora have been created. The popular corpora include TimeBank1.2, AQUAIN, TempEval, and TempEval2. Among them, the TempEval corpus, based on TimeBank1.2, was created for the temporal relation task at TempEval1 in 2007. For the Tempeval2 task in 2010, a multilingual corpus was created[112, 121]. Detailed information about these corpora can be found at http://www.timeml.org/site/timebank/timebank.html. Many TIE systems have been developed based on these available corpora [121].

In the general English domain, both machine learning and rule-based methods have been applied to TIE. Machine learning methods have been widely adopted, and demonstrated good performance on event extraction, including conditional random fields (CRFs) and supported vector machines (SVMs) [121]. For temporal expression extraction, both machine learning and rule-based methods were studied in TempEval2; in this test, rule-based methods slightly outperformed machine learning based methods [121]. All systems in TempEval2 identified temporal expressions attributes using rule-based methods [121]. HeidelTime, an open source system for temporal expression extraction, is a representative rule-based system that performed well in TempEval2 [122]. Temporal relation extraction is typically divided into different sub-tasks. For example, in TempEval2, TLinks were divided into three different types: (1) TLinks between event and documentation time; (2) TLinks between events/times within the same sentence; and (3) TLinks between events/times across sentences. Both machine learning based or rule-based methods were used for different sub-tasks in TempEval2. To date, performance of temporal relation extraction systems has been less than optimal—the best system in TimeEval2 competition achieved F-measures of 82%, 65%, and 58% on three types of TLinks [112]. More recently, researchers have investigated methods that can integrate constraints among TLinks from all sub-tasks to further improve TIE performance. For example, Naushad *et al* [123] used Markov Logic networks to model the constraints in all TLinks and showed improved performance.

Temporal information is crucial and important for many medical applications. A number of studies [124] have addressed various topics of temporal representation and reasoning with medical data. Processing temporal events in medical text, however, has not been extensively studied. A few studies have developed different methods to extract temporal expressions from clinical narratives [124, 125]. For example, Reeves *et al* extended the open-source temporal awareness and reasoning systems for question interpretation (TARSQI) toolkit, originally developed from news reports, to extract temporal expressions from Veterans Affairs (VA) clinical text. They found that temporal expressions in clinic notes were very different from those in the newswire domain, and the out-of-the-box implementation of the TARSQI toolkit performed poorly[126]. Some existing clinical NLP systems, such as ConText [127] and MedLEE [1], also have the capability to recognize certain temporal expressions and link them to clinical concepts. More comprehensive systems such as developed by Zhou *et al*[124, 128] can not only extract temporal expressions associated with medical events, but also reason about temporal information in clinical narrative reports. For more details of studies in clinical

TIE, see the review paper by Zhou and Hripcsak [129]. Nevertheless, very few studies have investigated the use of TimeML in the medical domain. Recent studies by Savova *et al* [2, 125] have annotated clinical text using TimeML.

Organizers of the 2012 i2b2 clinical NLP challenge organized a clinical TIE competition in order to advance the TIE research in the medical domain. The 2012 i2b2 challenge consisted of three subtasks: (1) *Event extraction*: six types of clinical events were extracted for the i2b2 challenge, including medical problems, tests, treatments, clinical departments, evidentials, and occurrences. Every event also has two attributes: polarity and modality. The polarity attribute marks whether an event is positive or negative, and the modality attribute is used to describe whether an event actually occurred or not. (2) *Temporal expression extraction*: the TIMEX3 tag was used to annotate temporal expressions, which has three main attributes: type (date/time/duration/frequency), value (normalized value of the TIMEX3), and modifier of a value (more, less, approximate, and so on). (3) *Temporal relation (TLink) extraction*: in this task, systems identified relations between events and times, and determined the type of relation. Three relation types (*before, overlap*, and *after*) were used in this challenge, as a simplification of the 13 more detailed ones specified in TimeML (simultaneous, before, after, immediately before, immediately after, including, being included, during, beginning, begun by, ending, identity, set/subset). All TLinks were further divided into three categories: (1) TLinks between events and section times (e.g., admission or discharge time); (2) TLinks between events/times within one sentence; and (3) TLinks between events/time across sentences (e.g., co-referenced entities).

The 2016 Clinical TempEval challenge is the most recent community challenge that addresses temporal information extraction from clinical notes. Following the 2015 Clinical TempEval challenge, the 2016 challenge consists of six sub-tasks, each of which is to identify: (1) spans of event mentions, (2) spans of time expressions, (3) attributes of events, (4) attribute of times, (5) events' temporal relations to the document creation times (DocTimeRel), and (6) narrative container relations among events and times. 440 annotated clinical notes from Mayo Clinic, or the THYME corpus (Styler IV et al., 2014), were provided as the training data set, and 153 plain text clinical notes were provided as the test set. The participating systems were evaluated through two phases. In phase 1, the systems were evaluated on their results for all six sub-tasks given plain texts as inputs. In phase 2, system predictions on DocTimeRel and TLINK:Contains were evaluated given the gold-standard event annotations (EVENT) and time annotations (TIMEX3).

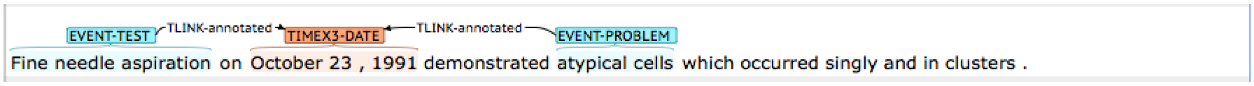
In most TIE systems, parsed tree of the sentences (constituency and dependency parsed tree) have been widely used as an important source to generate features for classifying temporal relations. In this study, we validated our parsers' improvement by applying them to a temporal information extraction task.

5.2.2 Methods

5.2.2.1 Dataset

The dataset used in this study is from the 2012 i2b2 clinical NLP challenge. In the challenge, 310 discharge summaries from Partners Healthcare and the Beth Israel Deaconess Medical Center were annotated for temporal information, including clinical event, temporal expression, and temporal relation. Temporal relations (TLINK) indicate whether and how two clinical events, two temporal expressions, or a clinical event and a temporal expression related to each other in the clinical timeline. Possible TLINK types include BEFORE, AFTER, SIMULTANEOUS, OVERLAP, BEGUN_BY, ENDED_BY, DURING, and BEFORE_OVERLAP. Identification of all such relations is a difficult task, as shown by the relatively low performance even of the top ranked system [130] in the challenge.

To simplify the task and make it more feasible, in this study, we removed TLINK types and limited the task to identify relations between TIMEX and EVENT within the same sentence only. Figure 11 shows an example annotation, where "Fine needle aspiration" and "atypical cells" are two clinical events and both of them are linked to temporal expression "October 23, 1991". We selected the 120 clinical notes from the test set of the challenge and re-annotated them according to these new criteria.



Fine needle aspiration on October 23 , 1991 demonstrated atypical cells which occurred singly and in clusters .

Figure 11. An example sentence of temporal relation annotation

5.2.2.3 Relation extraction systems

In this study, the task was to classify if a temporal expression is related to a clinical event, given the temporal expressions and clinical events in the sentence. We built a machine learning based classifier to solve this problem. The features we used included three categories: 1) Clinical event attributes; 2) Temporal expression attributes; 3) Dependency related features. Table 16 describes the complete feature set. For clarity, Table 17 lists the specific features extracted from the example sentence shown in Figure

11. As for the classification algorithm, we used the open source toolkit, Lib-linear, as an implementation of the SVM algorithm. For each implemented method, all parameters were tuned for optimal performance via cross-validation.

Table 16. Feature list of temporal relation extraction

Feature Group	**Description**
Clinical events attributes	
Tag of event	Problem, treatment or test
Temporal expression attributes	
TIMEX CLASS	Class of TIMEX including date, time, duration etc.
Dependency feature	
Path	Terminal nodes in the path of linking the clinical event and the temporal expression in the dependency parsed tree
Ngram of Path	1gram and 2gram of Path
Path POS	Part-of-speech tags in the path of linking the clinical event and the temporal expression in the dependency parsed tree
Ngram of Path POS	1gram and 2gram of Path POS

Table 17. An example of features extracted for temporal relation extraction

Sentence	Fine needle aspiration on October 23, 1991 demonstrated atypical cells which occurred singly and in clusters
Feature Group	Description
Clinical events attributes	
Tag of event	Type of clinical events including problem, treatment or test
Temporal expression attributes	
TIMEX CLASS	Class of TIMEX including date, time, duration etc.
Dependency	

feature	
Path	Terminal nodes in the path of linking the clinical event and the temporal expression in the dependency parsed tree
Ngram of Path	1gram and 2gram of Path
Path POS	Part-of-speech tags in the path of linking the clinical event and the temporal expression in the dependency parsed tree
Ngram of Path POS	1gram and 2gram of Path POS

5.2.2.4 Experiments

We examined two widely used state-of-the-art syntactic parsers, the Stanford parser [38] and the Berkley parser [39] for their influence on the temporal relation extraction task, since these ranked as the top two parsers among all the four state-of-the-parsers that we studied in the Chapter 3. Using 5-fold cross validation for evaluation, we compared different parsers in following settings:

1) Evaluate temporal relation extraction performance based on the parsers with their default settings: In this experiment, we directly applied two parsers to process all sentences of the dataset. Both parsers were invoked with their default settings and models, which had been trained on the Penn Treebank.

2) Evaluate temporal relation extraction performance based on parsers re-trained on the clinical Treebank: We applied both parsers retrained on the MiPACQ Treebank, to process all sentences in the dataset.

3) Evaluate temporal relation extraction performance based on parsers re-trained on the clinical treebank and integrated with our PP attachment disambiguation model: To assess if PP attachment disambiguation could further improve the performance of temporal relation extraction, we applied our PP disambiguation approach described in Chapter 4 to the retrained parsers and processed all sentences in the dataset.

5.2.3 Results

Table 18 shows the performance of temporal relation extraction systems based on different parsers and settings. For the Berkeley parser, temporal relation extraction system achieved an F-1 score of 78.95%, when features generated from the default setting were used. The F-1 score was improved to 80.45% after using the improved Berkeley parser to generate features. For the Stanford parser, those two numbers were 78.62% and 79.36%, respectively.

Table 18. Two categories temporal relation classifier performance based on the Stanford parser and Berkeley parser

Parser	Mode	Precision	Recall	F-1
Berkeley	Default	81.08%	76.93%	78.95%
	Retrained	83.27%	77.28%	80.16%
	Retrained + improved PP attachment	83.48%	77.63%	80.45%
Stanford	Default	81.08%	76.31%	78.62%
	Retrained	81.72%	76.15%	78.83%
	Retrained + improved PP attachment	81.99%	76.89%	79.36%

5.2.4 Discussion

Syntactic information from parse trees such as dependency relations is important for relation extraction tasks. In this study, we demonstrated the effectiveness of improved clinical parsers on temporal relation information extraction task. For both the Stanford parser and the Berkeley parser, we showed that by using the methods we introduced in Chapters 3 and 4, we can improve parsing performance, thus improving temporal relation extraction performance (an increase of 1.5% on F-1 score for the Berkeley parser and an increment of 0.74% for the Stanford Parser). These results demonstrate the importance of syntactic parsing and its utility for other NLP tasks in the medical domain.

5.2.5 Conclusion

In this section, we validated the effectiveness of optimized parser for the temporal relation extraction task. The result showed that using the feature generated by optimized parsers could further improve the performance of temporal relation extraction.

Chapter 6. Conclusion

6.1 Summary of key findings

In this research, I systematically investigated different approaches to improve syntactic parsing of clinical text. The key findings from each chapter are summarized in the following paragraphs.

In Chapter 1, I introduced syntactic parsing as a possible solution to address the challenge of information extraction in clinical studies. A literature review was conducted to show that state-of-the-art parsers have achieved great success in the open domain and researchers have also done some work on the evaluation of state-of-the-art parsers on both biomedical literature and clinical text. However, there is no comprehensive study that focuses on improving syntactic parsing using clinical domain knowledge. I also reviewed the literature on resolving PP and coordination ambiguities, which are important for improving parsing in the medical domain.

In Chapter 2, I developed annotation guidelines for annotating parse trees of clinical sentences and built two clinical treebanks: one for progress notes and one for discharge summaries. Our annotation guidelines address several unique challenges for annotating clinical sentences, including missing elements and superfluous and redundant elements. The two clinical treebanks from this study serve as a great resource to train and evaluate existing syntactic parsers, thus making it easy to adapt them to the medical domain.

In Chapter 3, I investigated the performance of four state-of-the-art parsers using different treebanks including both a general English treebank and several clinical treebanks. I found that retraining using clinical treebanks could greatly improve the performance of all parsers, just as expected. Among various experiments, the Berkeley parser achieved the best performance (an F-measure of 86.39%) when it was retrained on the MiPACQ corpus, which is comparable to its performance in the open domain. According to the error analysis, I found that a significant amount of errors were caused by the ambiguity of PP attachment and coordination.

In Chapter 4, I leveraged semantic information generated by existing high-performance clinical information extraction tools to resolve the ambiguities of PP attachment and coordination in the parse trees. For PP attachment ambiguity, I built a machine learning based solution to automatically identify ambiguous PP attachments between verb and noun. After using semantic and other features, there are 2.35% and 1.77% increases in accuracy for identifying PP attachments in the MiPACQ treebank and the DischargeSummaries treebank, respectively. However, the improvement on the overall parsing performance was limited. For coordination ambiguity, I developed a rule-based system to identify one certain type of coordination error and used top n-best parse trees to fix identified errors.

In Chapter 5, I applied the improved parsers to two clinical NLP tasks: SRL and temporal relation extraction. Our results show that improved syntactic parsers could increase the SRL performance significantly (with an increase of F-measure from 3.31% to 9.0%). For the temporal relation extraction task, syntactic features from the improved Berkeley parser also significantly increased TIE systems' performance (1.50% of F-1 score). These external validations demonstrate the importance and value of syntactic parsing approaches in other NLP applications in the medical domain.

6.2 Innovations and contributions

6.2.1 Innovations

To the best of our knowledge, this is one of the first comprehensive studies on syntactic parsing of clinical text. I have addressed several important aspects of parsing in the medical domain, ranging from guideline and Treebank development to new methods handling PP attachment and coordination ambiguities in clinical text. More specifically, this work is innovative in following aspects:

(1) New annotation guidelines for clinical treebanks were developed, with the capability to handle unique challenges presented in clinical text, such as missing elements, redundant constituents etc., which are not covered by the existing annotation guidelines for building general English treebanks.

(2) Following the annotation guidelines, two clinical treebanks were developed. They are among the first few clinical treebanks available for the community.

(3) I performed the first comprehensive study to investigate state-of-the-art parsers' performance on multiple types of clinical corpora.

(4) I developed a machine learning based method to resolve the PP attachment ambiguity and a rule-based method to help detect coordination ambiguity. Both methods are new to syntactic parsing of clinical text.

6.2.2 Contributions

This dissertation study contributed to the areas of biomedical informatics, computer science, and healthcare. The major contribution of this dissertation work to biomedical informatics is that it develops a framework for syntactic parsing of clinical text, by 1) creating annotated guideline and corpora; 2) investigating the performance of state-of-the-art parsers; 3) resolving PP attachment and coordination ambiguities in the parse trees; and 4) validating the effectiveness of optimized parser on semantic role labeling and temporal relation extraction. These will be valuable resources for method development in syntactic parsing on the clinical text. The initial attempt of resolving ambiguities in the parse trees provides insight for other researchers in the field. This study also contributes to computer and information science. It demonstrates that retraining on domain specific treebank is effective in closed domains such as medicine. Furthermore, this study also benefits healthcare. It proves the effectiveness of optimized parsers on other NLP tasks, thus making it possible to utilize optimized parsers to facilitate healthcare operation and clinical research.

6.3 Future work

Although we have conducted multiple studies of syntactic parsing in the medical domain, there is a long way to go in order to achieve desirable performance in parsing clinical text. Several approaches that were proposed here are limited to particular problems in syntactic paring of clinical text. For example, for PP attachment and coordination ambiguities, we only proposed methods for very limited types of ambiguity. Much work is needed in order to completely resolve ambiguities in syntactic parsing. Furthermore, we conducted experiments on limited types of clinical notes including progress notes, discharge summaries and pathology notes. In the future, we plan to extend our studies to other types of clinical notes such as operative notes, to assess the generalizability of our methods. Another interesting research direction is to investigate more advanced algorithms for parsing, e.g., new deep learning architectures for syntactic parsing.

6.4 Conclusion

In this dissertation research, I systematically studied how to leverage clinical domain knowledge to improve the performance of state-of-the-art parsers on clinical corpora. The experimental results showed that the proposed methods remarkably improved state-of-the-art parsers' performance on clinical text, which subsequently improved the performance of other clinical NLP tasks. To the best of my knowledge, this is the first comprehensive study on syntactic parsing in the medical domain, building a solid basis for further research and applications.

Reference

1. Friedman, C., *Towards a comprehensive medical language processing system: methods and issues.* Proc AMIA Annu Fall Symp, 1997: p. 595-9.
2. Savova, G.K., et al., *Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications.* J Am Med Inform Assoc, 2010. **17**(5): p. 507-13.
3. Xu, H., et al., *MedEx: a medication information extraction system for clinical narratives.* J Am Med Inform Assoc, 2010. **17**(1): p. 19-24.
4. Aronson, A.R., *Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program.* Proc AMIA Symp, 2001: p. 17-21.
5. Singh, A., et al., *Incorporating temporal EHR data in predictive models for risk stratification of renal function deterioration.* Journal of biomedical informatics, 2015. **53**: p. 220-228.
6. Tao, C., et al. *Towards event sequence representation, reasoning and visualization for EHR data*. in *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*. 2012. ACM.
7. Zhao, J., et al. *Handling temporality of clinical events for drug safety surveillance*. in *AMIA Annual Symposium Proceedings*. 2015. American Medical Informatics Association.
8. Friedman, C., P. Kra, and A. Rzhetsky, *Two biomedical sublanguages: a description based on the theories of Zellig Harris.* Journal of biomedical informatics, 2002. **35**(4): p. 222-235.
9. Gottesman, O., et al., *The electronic medical records and genomics (eMERGE) network: past, present, and future.* Genetics in Medicine, 2013. **15**(10): p. 761-771.
10. Xu, H., et al. *Extracting and integrating data from entire electronic health records for detecting colorectal cancer cases*. in *AMIA Annu Symp Proc*. 2011.
11. Demner-Fushman, D., W.W. Chapman, and C.J. McDonald, *What can natural language processing do for clinical decision support?* Journal of biomedical informatics, 2009. **42**(5): p. 760-772.
12. Xu, H., et al., *Validating drug repurposing signals using electronic health records: a case study of metformin associated with reduced cancer mortality.* Journal of the American Medical Informatics Association, 2014: p. amiajnl-2014-002649.
13. de Bruijn, B., et al., *Machine-learned solutions for three stages of clinical information extraction: the state of the art at i2b2 2010.* Journal of the American Medical Informatics Association, 2011. **18**(5): p. 557-562.
14. Jiang, M., et al., *A study of machine-learning-based approaches to extract clinical entities and their assertions from discharge summaries.* Journal of the American Medical Informatics Association, 2011. **18**(5): p. 601-606.
15. Roberts, K. and S.M. Harabagiu, *A flexible framework for deriving assertions from electronic medical records.* Journal of the American Medical Informatics Association, 2011. **18**(5): p. 568-573.

16. Mitchell, B., *Prepositional phrase attachment using machine learning algorithms*, 2004, University of Sheffield;.

17. Denny, J.C., et al., *"Understanding" Medical School Curriculum Content Using KnowledgeMap.* Journal of the American Medical Informatics Association, 2003. **10**(4): p. 351-362.

18. Jain, N.L., et al. *Identification of suspected tuberculosis patients based on natural language processing of chest radiograph reports*. in *Proceedings of the AMIA Annual Fall Symposium*. 1996. American Medical Informatics Association.

19. Jain, N.L. and C. Friedman. *Identification of findings suspicious for breast cancer based on natural language processing of mammogram reports*. in *Proceedings of the AMIA Annual Fall Symposium*. 1997. American Medical Informatics Association.

20. Melton, G.B. and G. Hripcsak, *Automated detection of adverse events using natural language processing of discharge summaries.* Journal of the American Medical Informatics Association, 2005. **12**(4): p. 448-457.

21. Friedman, C., et al., *Automated encoding of clinical documents based on natural language processing.* Journal of the American Medical Informatics Association, 2004. **11**(5): p. 392-402.

22. Xu, H., et al., *Facilitating cancer research using natural language processing of pathology reports.* Studies in health technology and informatics, 2003. **107**(Pt 1): p. 565-572.

23. Denny, J.C., et al. *Identifying UMLS concepts from ECG Impressions using Knowledge Map*. in *AMIA*. 2005.

24. Denny, J.C., et al., *Evaluation of a method to identify and categorize section headers in clinical documents.* Journal of the American Medical Informatics Association, 2009. **16**(6): p. 806-815.

25. Mitchell, K.J., et al., *Implementation and evaluation of a negation tagger in a pipeline-based system for information extraction from pathology reports.* Medinfo, 2004. **2004**: p. 663-7.

26. Mehrabi, S., et al., *DEEPEN: A negation detection system for clinical text incorporating dependency relation into NegEx.* Journal of biomedical informatics, 2015. **54**: p. 213-219.

27. Sohn, S., et al., *MedXN: an open source medication extraction and normalization tool for clinical text.* Journal of the American Medical Informatics Association, 2014. **21**(5): p. 858-865.

28. Bies, A., et al., *Bracketing guidelines for treebank ii style penn treebank project, 1995.* URL *http://www. cis. upenn. edu/~* treebank, 1995.

29. Gildea, D. and M. Palmer, *The necessity of parsing for predicate argument recognition*, in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*2002, Association for Computational Linguistics: Philadelphia, Pennsylvania. p. 239-246.

30. Callison-Burch, C., *Syntactic constraints on paraphrases extracted from parallel corpora*, in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*2008, Association for Computational Linguistics: Honolulu, Hawaii. p. 196-205.

31. Proudian, D. and C. Pollard. *Parsing head-driven phrase structure grammar*. in *Proceedings of the 23rd annual meeting on Association for Computational Linguistics*. 1985. Association for Computational Linguistics.

32. Hindle, D. *Deterministic parsing of syntactic non-fluencies*. in *Proceedings of the 21st annual meeting on Association for Computational Linguistics*. 1983. Association for Computational Linguistics.

33. Magerman, D.M., *Statistical decision-tree models for parsing*, in *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*1995, Association for Computational Linguistics: Cambridge, Massachusetts.

34. Collins, M., *Three generative, lexicalised models for statistical parsing*, in *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*1997, Association for Computational Linguistics: Madrid, Spain.

35. Bikel, D.M., *On the parameter space of generative lexicalized statistical parsing models.* 2004.

36. Charniak, E. and M. Johnson, *Coarse-to-fine <i>n</i>-best parsing and MaxEnt discriminative reranking*, in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*2005, Association for Computational Linguistics: Ann Arbor, Michigan.

37. McClosky, D., E. Charniak, and M. Johnson, *Effective self-training for parsing*, in *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*2006, Association for Computational Linguistics: New York, New York.

38. Klein, D. and C.D. Manning, *Accurate unlexicalized parsing*, in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*2003, Association for Computational Linguistics: Sapporo, Japan. p. 423-430.

39. Petrov, S. and D. Klein, *Improved Inference for Unlexicalized Parsing.* 2007.

40. Collobert, R., *Deep Learning for Efficient Discriminative Parsing.* JMLR Proceedings, 2011. **15**: p. 8.

41. Richard Socher, J.B., Christopher D Manning, Andrew Y Ng, *Parsing with compositional vector grammars.* Proceedings of the ACL conference, 2013.

42. Lease, M. and C. E. *Parsing Biomedical Literature*. in *the Second International Joint Conference on Natural Language Processing (IJCNLP'05)*. 2005. Jeju Island, Korea.

43. Tateisi, Y., et al., *The genia corpus: Medline abstracts annotated with linguistic information.* Third meeting of SIG on Text Mining, Intelligent Systems for Molecular Biology (ISMB), 2003.

44. Clegg, A.B. and A.J. Shepherd, *Benchmarking natural-language parsers for biological applications using dependency graphs.* BMC Bioinformatics, 2007. **8**: p. 24.

45. Sager, N., et al. *The analysis and processing of clinical narrative.* in *MedInfo*. 1986.

46. Sager, N., C. Friedman, and M. Lyman, *Medical language processing: computer management of narrative data.* 1987, Reading, MA: Addison-Wesley.

47. Friedman, C., P. Kra, and A. Rzhetsky, *Two biomedical sublanguages: a description based on the theories of Zellig Harris.* J Biomed Inform, 2002. **35**(4): p. 222-35.

48. Friedman, C., et al., *A general natural-language text processor for clinical radiology.* J Am Med Inform Assoc, 1994. **1**(2): p. 161-74.

49. Hripcsak, G., et al., *Use of natural language processing to translate clinical information from a database of 889,921 chest radiographic reports.* Radiology, 2002. **224**(1): p. 157-63.

50. Haug, P.J., et al., *Experience with a mixed semantic/syntactic parser.* Proc Annu Symp Comput Appl Med Care, 1995: p. 284-8.

51. Haug, P.J., et al., *A natural language parsing system for encoding admitting diagnoses.* Proc AMIA Annu Fall Symp, 1997: p. 814-8.

52. Huang, Y., et al., *Improved identification of noun phrases in clinical radiology reports using a high-performance statistical natural language parser augmented with the UMLS specialist lexicon.* J Am Med Inform Assoc, 2005. **12**(3): p. 275-85.

53. Albright, D., et al., *Towards comprehensive syntactic and semantic annotations of the clinical narrative.* Journal of the American Medical Informatics Association, 2013.

54. Wang, Y., et al., *Domain adaption of parsing for operative notes.* Journal of biomedical informatics, 2015. **54**: p. 1-9.

55. Volk, M. *Exploiting the WWW as a corpus to resolve PP attachment ambiguities*. in *Proceedings of Corpus Linguistics*. 2001.

56. Collins, M. and J. Brooks, *Prepositional phrase attachment through a backed-off model.* arXiv preprint cmp-lg/9506021, 1995.

57. Stetina, J. and M. Nagao. *Corpus based PP attachment ambiguity resolution with a semantic dictionary*. in *Proceedings of the fifth workshop on very large corpora*. 1997. Citeseer.

58. Toutanova, K., et al. *Parse disambiguation for a rich HPSG grammar*. in *First Workshop on Treebanks and Linguistic Theories (TLT2002), 253-263*. 2002. Stanford InfoLab.

59. Hindle, D. and M. Rooth, *Structural ambiguity and lexical relations.* Computational linguistics, 1993. **19**(1): p. 103-120.

60. Ratnaparkhi, A., *Maximum entropy models for natural language ambiguity resolution*, 1998, University of Pennsylvania.

61. Resnik, P., *Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language.* J. Artif. Intell. Res.(JAIR), 1999. **11**: p. 95-130.

62. Nakov, P. and M. Hearst. *Using the web as an implicit training set: application to structural ambiguity resolution*. in *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*. 2005. Association for Computational Linguistics.

63. Goldberg, M. *An unsupervised model for statistically determining coordinate phrase attachment*. in *Proceedings of the 37th annual meeting of the*

Association for Computational Linguistics on Computational Linguistics. 1999. Association for Computational Linguistics.

64. Chapman, W.W., et al., *Overcoming barriers to NLP for clinical text: the role of shared tasks and the need for additional creative solutions.* Journal of the American Medical Informatics Association, 2011. **18**(5): p. 540-543.
65. Marcus, M.P., M.A. Marcinkiewicz, and B. Santorini, *Building a large annotated corpus of English: the penn treebank.* Comput. Linguist., 1993. **19**(2): p. 313-330.
66. Uzuner, O., *Recognizing obesity and comorbidities in sparse data.* J Am Med Inform Assoc, 2009. **16**(4): p. 561-70.
67. Uzuner, O., I. Solti, and E. Cadag, *Extracting medication information from clinical text.* J Am Med Inform Assoc, 2010. **17**(5): p. 514-8.
68. Uzuner, Ö., et al., *2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text.* Journal of the American Medical Informatics Association, 2011.
69. Rea, S., et al., *Building a robust, scalable and standards-driven infrastructure for secondary use of EHR data: the SHARPn project.* Journal of biomedical informatics, 2012. **45**(4): p. 763-771.
70. Warner, C., et al., *Bracketing biomedical text: an addendum to Penn Treebank II guidelines.* 2012.
71. Foster, J., *Treebanks gone bad.* International Journal of Document Analysis and Recognition (IJDAR), 2007. **10**(3-4): p. 129-145.
72. Morton T, L.J., *an open tool for linguistic annotation.* Proc Conf N Am Chap Assoc Comput Linguist Hum Lang Technol., 2003. **17-8**.
73. Socher, R., et al. *Parsing with Compositional Vector Grammars*. in *ACL (1)*. 2013.
74. Saeed, M., et al., *Multiparameter Intelligent Monitoring in Intensive Care II (MIMIC-II): a public-access intensive care unit database.* Critical care medicine, 2011. **39**(5): p. 952.
75. Ratnaparkhi, A., J. Reynar, and S. Roukos. *A maximum entropy model for prepositional phrase attachment*. in *Proceedings of the workshop on Human Language Technology*. 1994. Association for Computational Linguistics.
76. Porter, M.F., *Snowball: A language for stemming algorithms*, 2001.
77. Fan, R.-E., et al., *LIBLINEAR: A library for large linear classification.* Journal of machine learning research, 2008. **9**(Aug): p. 1871-1874.
78. Xu, J., et al., *UTH-CCB: the participation of the SemEval 2015 challenge–Task 14.* Proceedings of SemEval-2015, 2015.
79. Liu, Y., et al. *Using SemRep to label semantic relations extracted from clinical text.* in *AMIA*. 2012.
80. Rindflesch, T.C. and M. Fiszman, *The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text.* Journal of biomedical informatics, 2003. **36**(6): p. 462-477.
81. Cohen, K.B., M. Palmer, and L. Hunter, *Nominalization and alternations in biomedical language.* PloS one, 2008. **3**(9): p. e3158.

82. Pradhan, S.S., et al. *Shallow Semantic Parsing using Support Vector Machines*. in *HLT-NAACL*. 2004.

83. Palmer, M., D. Gildea, and P. Kingsbury, *The proposition bank: An annotated corpus of semantic roles.* Computational linguistics, 2005. **31**(1): p. 71-106.

84. Shen, D. and M. Lapata. *Using Semantic Roles to Improve Question Answering*. in *EMNLP-CoNLL*. 2007.

85. Zhang, R., et al., *Coherent narrative summarization with a cognitive model.* Computer Speech & Language, 2016. **35**: p. 134-160.

86. Moschitti, A., et al. *Exploiting syntactic and shallow semantic kernels for question answer classification*. in *Annual meeting-association for computational linguistics*. 2007.

87. Luo, Y., Ö. Uzuner, and P. Szolovits, *Bridging semantics and syntax with graph algorithms—state-of-the-art of extracting biomedical relations.* Briefings in bioinformatics, 2017. **18**(1): p. 160-178.

88. Surdeanu, M., et al. *The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies*. in *Proceedings of the Twelfth Conference on Computational Natural Language Learning*. 2008. Association for Computational Linguistics.

89. Wattarujeekrit, T., P.K. Shah, and N. Collier, *PASBio: predicate-argument structures for event extraction in molecular biology.* BMC bioinformatics, 2004. **5**(1): p. 155.

90. Bethard, S., et al., *Semantic role labeling for protein transport predicates.* BMC bioinformatics, 2008. **9**(1): p. 277.

91. D'Souza, J. and V. Ng, *Classifying temporal relations in clinical data: a hybrid, knowledge-rich approach.* Journal of biomedical informatics, 2013. **46**: p. S29-S39.

92. Kingsbury, P. and M. Palmer. *From TreeBank to PropBank*. in *LREC*. 2002. Citeseer.

93. Charniak, E. and M. Johnson. *Coarse-to-fine n-best parsing and MaxEnt discriminative reranking*. in *Proceedings of the 43rd annual meeting on association for computational linguistics*. 2005. Association for Computational Linguistics.

94. Xue, N. and M. Palmer. *Calibrating Features for Semantic Role Labeling*. in *EMNLP*. 2004.

95. Merlo, P. and G. Musillo. *Semantic parsing for high-precision semantic role labelling*. in *Proceedings of the Twelfth Conference on Computational Natural Language Learning*. 2008. Association for Computational Linguistics.

96. Meza-Ruiz, I. and S. Riedel. *Jointly identifying predicates, arguments and senses using Markov logic*. in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. 2009. Association for Computational Linguistics.

97. Huang, Y., et al., *Improved identification of noun phrases in clinical radiology reports using a high-performance statistical natural language parser augmented with the UMLS specialist lexicon.* Journal of the American Medical Informatics Association, 2005. **12**(3): p. 275-285.

98. Fan, J.-w., et al., *Syntactic parsing of clinical text: guideline and corpus development with handling ill-formed sentences.* Journal of the American Medical Informatics Association, 2013. **20**(6): p. 1168-1177.

99. Wang, Y., S. Pakhomov, and G.B. Melton, *Predicate argument structure frames for modeling information in operative notes.* Studies in health technology and informatics, 2013. **192**: p. 783.

100. Styler IV, W.F., et al., *Temporal annotation in the clinical domain.* Transactions of the Association for Computational Linguistics, 2014. **2**: p. 143-154.

101. Punyakanok, V., D. Roth, and W.-t. Yih, *The importance of syntactic parsing and inference in semantic role labeling.* Computational Linguistics, 2008. **34**(2): p. 257-287.

102. Zhang, Y., et al., *Domain adaptation for semantic role labeling of clinical text.* Journal of the American Medical Informatics Association, 2015: p. ocu048.

103. De Marneffe, M.-C. and C.D. Manning. *The Stanford typed dependencies representation*. in *Coling 2008: proceedings of the workshop on cross-framework and cross-domain parser evaluation*. 2008. Association for Computational Linguistics.

104. Pradhan, S., et al., *Support vector learning for semantic argument classification.* Machine Learning, 2005. **60**(1-3): p. 11-39.

105. Dahlmeier, D. and H.T. Ng, *Domain adaptation for semantic role labeling in the biomedical domain.* Bioinformatics, 2010. **26**(8): p. 1098-1104.

106. Tsai, R.T.-H., et al., *BIOSMILE: a semantic role labeling system for biomedical verbs using a maximum-entropy model with automatically generated template features.* BMC bioinformatics, 2007. **8**(1): p. 325.

107. Gildea, D. and D. Jurafsky, *Automatic labeling of semantic roles.* Computational linguistics, 2002. **28**(3): p. 245-288.

108. Alonso, O., et al., *Temporal Information Retrieval: Challenges and Opportunities.* Twaw, 2011. **11**: p. 1-8.

109. Goldstein, J., et al. *Multi-document summarization by sentence extraction*. in *Proceedings of the 2000 NAACL-ANLPWorkshop on Automatic summarization-Volume 4*. 2000. Association for Computational Linguistics.

110. Pustejovsky, J., et al., *TimeML: Robust specification of event and temporal expressions in text.* New directions in question answering, 2003. **3**: p. 28-34.

111. Verhagen, M., et al., *The TempEval challenge: identifying temporal relations in text.* Language Resources and Evaluation, 2009. **43**(2): p. 161-179.

112. Verhagen, M., et al. *SemEval-2010 task 13: TempEval-2*. in *Proceedings of the 5th international workshop on semantic evaluation*. 2010. Association for Computational Linguistics.

113. Allen, J.F., *Maintaining knowledge about temporal intervals.* Communications of the ACM, 1983. **26**(11): p. 832-843.

114. Sundheim, B.M. *Overview of results of the MUC-6 evaluation*. in *Proceedings of a workshop on held at Vienna, Virginia: May 6-8, 1996*. 1996. Association for Computational Linguistics.

115. Chinchor, N.A., *Overview of muc-7/met-2.* 1998.

116. Mani, I., et al. *Guidelines for annotating temporal information*. in *Proceedings of the first international conference on Human language technology research*. 2001. Association for Computational Linguistics.

117. Ferro, L., et al. *Annotating temporal information: from theory to practice*. in *Proceedings of the second international conference on Human Language Technology Research*. 2002. Morgan Kaufmann Publishers Inc.

118. Doddington, G.R., et al. *The Automatic Content Extraction (ACE) Program-Tasks, Data, and Evaluation*. in *LREC*. 2004.

119. Setzer, A., *Temporal information in newswire articles: an annotation scheme and corpus study*, 2002, University of Sheffield.

120. Setzer, A. and R. Gaizauskas. *A pilot study on annotating temporal relations in text*. in *Proceedings of the workshop on Temporal and spatial information processing-Volume 13*. 2001. Association for Computational Linguistics.

121. Pustejovsky, J. and M. Verhagen. *SemEval-2010 task 13: evaluating events, time expressions, and temporal relations (TempEval-2)*. in *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*. 2009. Association for Computational Linguistics.

122. Strötgen, J. and M. Gertz. *Heideltime: High quality rule-based extraction and normalization of temporal expressions*. in *Proceedings of the 5th International Workshop on Semantic Evaluation*. 2010. Association for Computational Linguistics.

123. UzZaman, N. and J.F. Allen. *TRIPS and TRIOS system for TempEval-2: Extracting temporal information from text*. in *Proceedings of the 5th International Workshop on Semantic Evaluation*. 2010. Association for Computational Linguistics.

124. Zhou, L., et al. *System Architecture for Temporal Information Extraction, Representationand Reasoning in Clinical Narrative Reports*. in *AMIA*. 2005.

125. Savova, G., et al. *Towards temporal relation discovery from the clinical narrative*. in *AMIA*. 2009.

126. Reeves, R.M., et al., *Detecting temporal expressions in medical narratives.* International journal of medical informatics, 2013. **82**(2): p. 118-127.

127. Chapman, W.W., D. Chu, and J.N. Dowling. *ConText: An algorithm for identifying contextual features from clinical text*. in *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*. 2007. Association for Computational Linguistics.

128. Zhou, L., et al., *A temporal constraint structure for extracting temporal information from clinical narrative.* Journal of biomedical informatics, 2006. **39**(4): p. 424-439.

129. Zhou, L. and G. Hripcsak, *Temporal reasoning with medical data—a review with emphasis on medical natural language processing.* Journal of biomedical informatics, 2007. **40**(2): p. 183-202.

130. Sun, W., A. Rumshisky, and O. Uzuner, *Evaluating temporal relations in clinical text: 2012 i2b2 Challenge.* Journal of the American Medical Informatics Association, 2013. **20**(5): p. 806-813.