A STUDY OF RELATION EXTRACTION FOR BIOMEDICAL TEXT

by

Yifan Peng

A dissertation submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Science

Summer 2016

A STUDY OF RELATION EXTRACTION

FOR BIOMEDICAL TEXT

by

Yifan Peng

Approved: _____

Kathleen F. McCoy, Ph.D.
Chair of the Department of Computer and Information Sciences

Approved: _____

Babatunde A. Ogunnaike, Ph.D.
Dean of the College of Engineering

Approved: _____

Ann L. Ardis, Ph.D.
Senior Vice Provost for Graduate and Professional Education

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Vijay K. Shanker, Ph.D.
Professor in charge of dissertation

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Cathy H. Wu, Ph.D.
Professor in charge of dissertation

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Li Liao, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Zhiyong Lu, Ph.D.
Member of dissertation committee

ACKNOWLEDGEMENTS

Foremost, I would like to express my sincere gratitude to my advisor Prof. Cathy H. Wu and K. Vijay-Shanker for the continuous support of my Ph.D study and research, for their patience, motivation, enthusiasm, and immense knowledge. Their guidance helped me in all the time of research and writing of this thesis. I could not have imagined having better advisors and mentor for my Ph.D study.

Besides my advisors, I would like to thank the rest of my thesis committee: Prof. Liao Li and Dr. Zhiyong Lu, for their encouragement, insightful comments, and hard questions. My sincere thanks also goes to Dr. Zhiyong Lu for offering me the summer internship opportunities in his group and leading me working on diverse exciting projects.

Special thanks go to my fellow labmates in University of Delaware BioNLP Group: Ruoyao Ding, Samir Gupta, Gang Li, and Ashique Mahmood, for the stimulating discussions, for the days we were working together before deadlines, and for all the fun we have had in the last six years. Also I would like to thank my colleagues in National Center for Biotechnology Information, Chih-Hsuan Wei, Edwin Huang, and Robert Leaman, who have provided a stimulating environment and were always available when I needed to discuss ideas.

I would like to thank Xiaoran Wang and Yuanfang Chen for their friendship and for their encouragements at key moments throughout my graduate studies.

I gratefully acknowledge Ziyang Xu for providing feedback for my thesis and for playing devil's advocate so I could find out any flaws in my proposition.

Lastly, and most importantly, I wish to thank my parents for raising and loving me unconditionally, for encouraging me to explore the world, and for supporting me spiritually throughout my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

A crucial area of Biomedical Natural Language Processing is relation extraction, the study of identifying relations between entities. One main challenge of relation extraction is text variations. They hinder pattern-based approaches to encode all patterns necessary for achieving a high recall, and limit the generalizability of machine-learning models especially when the size of training data is small.

This thesis exams the representation of sentences for relation extraction. In particular, we are concerned with a suitable level of abstraction, which will improve the performance of the relation extraction systems, and in turn lead to advances in other text-mining fields.

This thesis describes three steps along these lines. First, we propose an automatic approach for sentence simplification. It reduces the sentence complexity by detecting various syntactic constructs and generating simplified sentences. Second, we describe a framework to facilitate the development of pattern-based biomedical relation extraction systems. The framework leverages various linguistic theories to semi-automatically generate lexico-syntactic patterns. It also applies sentence simplification and semantic relations to increase the pattern coverage. Finally, we propose a structured representation, called Extended Dependency Graph (EDG). It provides an abstract representation accounting for textual variations, by not only considering syntactic dependencies between words in a sentence, but also utilizing information beyond syntax to capture dependencies.

In each of these steps, we conduct experiments to evaluate the efficacy of the ideas. The results (1) show that various text-mining approaches can benefit from sentence simplification, (2) demonstrate that we can create state-of-the-art pattern-based systems using the framework to extract different types of relations, and (3) validate the utility of EDG in both pattern-based and machine-learning relation extraction systems.

Chapter 1

INTRODUCTION

Due to the accelerated growth of biomedical publications, it is becoming increasingly difficult for scientists to keep up with the new findings reported in the literature. As a consequence, there has been an increased effort to develop biomedical text mining tools and automatically extract information from the research literature. One of the common tasks in biomedical text mining is to extract binary relations between entities. While much progress has been developed during the last decade, there is still much room for improvement. Articles by nature (especially the abstracts) are dense with information and often use complex constructions. The amount of textual variations can thus be problematic for pattern-based and machine-learning methods. In light of this, my dissertation will provide insights on the representation of natural texts used in relation extraction task. We hope that advances in a suitable level of abstraction will improve the performance of the relation extraction systems, which in turn might lead to advances in other text mining fields.

In Section 1.1, we briefly discuss advantages and shortcomings of existing relation extraction methods. Section 1.2 presents the thesis contributions. Finally, Section 1.3 provides an outline of this thesis.

1.1 Relation Extraction

Relation extraction is a task to identify relations between entities mentioned in natural language texts. Most relation extraction systems focus on extracting binary relations. Examples include *protein-protein interaction* and *phosphorylation*. While current state-of-the-art named entities recognizers can automatically label data with high accuracy, the whole relation extraction process is still not a trivial task.

Approaches to the relation extraction task can be categorized into two major classes: (1) pattern-based approaches and (2) machine-learning approaches. Pattern-based systems often use hand-coded rules, therefore do not require annotated data to train a system. However, pattern-based systems require domain experts to be closely involved in the design and implementation of the system to capture the patterns used for extracting the necessary information. Some systems rely on extraction patterns defined at the surface textual level or shallow parsing [29, 47, 50, 91, 133]; others use deep parsers [42, 56, 63, 116]. In all these cases, rigid extraction patterns are manually encoded in the systems. Thus pattern-based approaches can usually achieve high precision. On the other hand, pattern-based approaches are often criticized to have low recall as well. This is most likely due to the fact that there are a large number of sentential variations in the text, and manually generating patterns is a labor- and time-intensive process which would require monetary investment as well. To address this problem, this dissertation contributes to the knowledge of the complexity of the texts in the biomedical domain. We unify different forms of syntactic variations to assist in reducing the number of patterns needed to develop relation extraction systems and overcoming the text complexity challenge.

Machine-learning systems treat the relation extraction task as a classification problem. Given a set of positive and negative relation examples, syntactic and semantic information can be extracted from the text. These two types of information then serve as cues for deciding whether the entities in a sentence are related or not. Machine-learning approaches are data-driven and can derive models for automated extraction from a set of annotated data [3, 8, 14, 65, 84, 114, 139]. The input to the classifier can be either a set of features extracted from sentences or rich structural representation like trees and graphs. Both can be used in discriminative classifiers such as maximum entropy classifiers [7], support vector machines [137], and conditional random fields [69]. However, machine-learning methods also meet challenges in coping with text variations. In the biomedical domain, this problem is more critical because the methods are limited by the small size of the training dataset as well. To address this problem, this dissertation strives to find a suitable level of abstraction in the text representation so that machine-learning methods become easier to generalize. Together with

the development of advanced kernel methods and the use of sophisticated parameter tuning in recent years, we show that the advances in both orthogonal directions can be combined to create favorably comparable relation extraction systems.

1.2 Thesis Contributions

The primary goal of this thesis is to develop a new representation by employing syntactic dependency information and, linguistic principles that go beyond syntax, and biomedical domain knowledge into a unified structure. We apply this representation to a number of biomedical relation extraction tasks and demonstrate that the resulting impact of our work is able to extend various tasks, such as protein-protein interaction, GENIA event, and mRNA-target relation. Performances of all methods presented in this thesis have been shown to be either comparable to or better than those of state-of-the-art systems. Our research contributions are as follows.

1. A sentence simplification system that reduces the syntactic complexity of sentence structures [101, 102, 104]. Sentence simplification is a technique to detect various types of clauses and constructs contributing to the complexity of sentences, and to produce two or more simple sentences while maintaining both coherence and the communicated message. By reducing the complexity, sentence simplification can ease the development of text-mining tools, as well as other NLP tools such as machine translation. For the purpose of demonstration, consider the following sentence,

Example 1 A third genetic linkage to disease is alpha-synuclein, a protein that is heavily phosphorylated in Lewy bodies and Lewy neuritis, the pathological hallmarks of PD.

Sentence simplification can detect different syntactic constructs from the sentence such as coordination ("Lewy bodies and Lewy neuritis"), relative clause ("a protein that is ...") and apposition ("alpha-synuclein, a protein that ..."). These constructs make major contributions to the sentence complexity. After the detection, sentence simplification can break this sentence into several simple ones, such as "Alpha-synuclein is heavily phosphorylated in Lewy bodies." and "Alpha-synuclein is heavily phosphorylated in Lewy neuritis." As can be seen, both

pattern-based or machine-learning approaches will undoubtedly find that it is much easier to process the simplified sentences than the original one.

In this dissertation, we present various types of simplification constructs that are frequently encountered in the biomedical literature and describe methods to detect and simplify them. We also conduct multiple experiments to show that sentence simplification can improve the coverage of extraction patterns and ease the difficulty of machine-learning methods in coping with text variations.

2. A generalizable NLP framework that can assist in developing pattern-based biomedical relation extraction systems [76, 103]. In our next step, we propose a framework to assist in developing pattern-based relation extraction systems. First, we leverage more syntactic variations possible in a language to automatically derive various patterns in a systematic manner. We then use sentence simplification to design a small set of patterns to match simple sentence constructs. We show that with the help of sentence simplification, we do not need to account for all complex syntactic constructs and generate an exhaustively large amount of patterns. Finally we identify referential relations to seek the most appropriate phrase referring to the target entity. Referential relations, by definition, are links between two phrases that refer to the same entity (e.g., coreference relation) or in a particular relation (e.g., part-of relation). By using these links, we can go beyond the patterns by searching from the syntactic argument of a predicate to the actual target. As a result, we are able to extract the target entities expected in the relation extraction task.

In this part of my dissertation, we designed and implemented a relation extraction system derived from the proposed framework. Then we evaluated the performance of the system on several biomedical relation corpora. We show that by taking the specification of trigger words only, we can produce a relation extraction system with results that compared favorably with state-of-the-art. We further used the derived system on more biomedical relation extraction tasks and more widely-used corpora. By providing a comprehensive and comparative analysis of various results, we demonstrate the generalizability of the proposed framework, which may play a role in switching the system effectively from one relation extraction task to another.

The fact that only the specification of the triggers is required from domain experts, together with the fact that no training set is required, meets our goals for developing the framework: ability to create effective relation extraction systems for new relations where resources (e.g., annotated corpus or database) are not publicly available.

3. A new representation, Extended Dependency Graph, that abstracts away text variations [105, 106]. Both the sentence simplification and the framework described above were developed by leveraging syntactic knowledge. We then assisted in developing the relation extraction systems by considering a new representation that goes beyond the syntax. The new representation, called Extended Dependency Graph (EDG), applies varied linguistic and domain knowledge to exploit semantic dependencies between entities. Our hypothesis is that EDG is suitable for extracting the biomedical relationships for both pattern-based and machine-learning systems, because it allows different text variations to have the same representation. To use EDG in the machine-learning system, we applied a simple edit distance kernel (edit kernel) and a more elaborate kernel (all-path graph kernel). We showed that superior performance can be achieved for five publicly available protein-protein interaction corpora using EDG. To use EDG in a rule-based system, we re-implemented the system based on the framework discussed in Chapter 4. Evaluation on the same corpus shows that we can achieve better results than the previous system which relies on the syntactic parse trees.

1.3 Thesis Overview

In Chapter 2, we review current approaches for sentence simplification, pattern-based and machine-learning relation extraction systems, and some widely-used corpora. Chapter 3 presents a sentence simplification system to reduce the syntactic complexity of sentence structure. In Chapter 4, we introduce a generalizable NLP framework that can be used to quickly develop pattern-based systems. Chapter 5 details a notable representation that manipulates both linguistic and semantic knowledge to interpret relations in the natural text. It further discusses results obtained by this new representation in both pattern-based and machine-learning system on different datasets. Chapter 6 summarizes this thesis and outlines directions for future work.

Chapter 2

RELATED WORK

This chapter introduces related work and concepts necessary for understanding the works presented in this thesis. We first discuss techniques of sentence simplification in Section 2.1 because it is a prerequisite for rule-based and machine-learning approaches to relation extraction and other tasks in general. Section 2.2 describes the relation extraction task, followed by an introduction to related work of pattern-based approaches in Section 2.2.1 and learning-based approaches in Section 2.2.2. Section 2.3 discusses several corpora in the biomedical domain that can be used for the evaluation.

2.1 Sentence Simplification

In Chapter 3, we proposed an alternative approach to detect and extract information from complex sentences. Instead of matching patterns to all possible variations in text, we propose to simplify complex sentences first, and then attempt to match simple patterns to the simplified sentences. The hypothesis is that sentence simplification can alleviate the problems of text mining tools when dealing with complexities [123].

Text simplification, defined narrowly, is the process of reducing the linguistic complexity of a text, while still retaining the original information content and meaning [122]. Automatic simplification of sentences was first introduced to improve the performance of systems which rely on natural language input [20, 21]. Later, a broader range of simplification approaches was proposed to help people with aphasia [18, 37], increase the readability of literature from college level to high school level [99], or improve the performance of natural language processing applications in various other areas [51, 64, 111, 138].

In the biomedical domain, various groups have used text simplification components as a pre-processing tool for relation extraction and text-mining applications [17, 54, 86, 99, 101].

Such applications rely on identifying lexico-syntactic patterns in text that express the semantic information to be mined. By simplifying complex sentences first, and then matching patterns in the simplified sentences, certain problems with data sparsity during pattern acquisition can be overcome.

Different levels of linguistic knowledge can be utilized when building simplification systems. Some use only word and phrase information to align from complex sentences to simplified sentences [32, 128, 141]. Others have evolved from research in syntax, such as shallow parsing [21, 101, 104, 121], synchronous tree-adjoining grammar [39], constituency-based parse trees [20, 54, 86, 140], and dependency-based parse trees [36, 105]. Besides syntax, morphological information is also used to handle voice change [124].

With regard to the evaluation, BioSimplify asked judges to evaluate the precision and recall by thinking of all possible grammatically correct simplified sentences of the original ones and reading each simplified sentence produced [54]. SimText compared the Flesch reading scores of sentences before and after simplification [99]. However, these methods do not report on the accuracy of the simplifier in detecting all possible simplification constructs in a sentence.

2.2 Relation Extraction in the Biomedical Domain

Relation extraction is a task to identify relations between entities mentioned in natural language texts. It can be further treated as two subtasks: whether there will be a relation between entities, and what is the type of that relation. Approaches to the relation extraction task can be categorized into two major classes: (1) pattern-based approaches and (2) machine learning-based approaches.

2.2.1 Pattern-based Approaches

Pattern-based approaches do not require annotated data to train a system. However, they do require domain experts to be closely involved in the design and implementation of the system to capture the patterns used for extracting the necessary information. Some systems rely on extraction patterns defined at the surface textual level or based on outputs

from a shallow parser [29, 47, 50, 91, 133]. Others use deep parsers with hand-crafted patterns [42, 56, 63, 116]. As found in OpenDMAP [53], a semantic grammar may be utilized with text literals, syntactic constituents, and semantic types of entities. Other notable rule-based systems include [1, 10, 72, 100, 131] based on link grammar [2, 107], dependency parsing [42, 116], and full parser [34, 142]. In all these cases, rigid extraction patterns are manually encoded in the systems. Because of the rigid patterns, pattern-based approaches usually achieve high precision but often have low recall. While it is feasible to manually identify and implement high-quality patterns to achieve good precision, it is often impractical to exhaustively encode all the patterns necessary for a high recall.

Among different pattern-based approaches, Kim and Rebholz-Schuhmann [63] used an HPSG parser to identify predicate-argument structure, and converted them into dependency structures. Syntactic patterns were defined on a dependency structure, and labeled with part-of-speech tags and named entities. To increase the recall of the system, they loosely matched the patterns by allowing to match a dependent item to any descendant of a node. Kilicoglu and Bergler [56] was a notable rule-based system proposed in BioNLP-ST 2011. They defined rules on embedded graphs, which were converted from syntactic dependency relations. These rules were collected based on most frequent dependency paths in training set. For event extraction, they use simple entities and triggers to help pattern matching. Narayanaswamy et al. [91] used manually developed patterns to extract phosphorylation information from text. Patterns were designed on base-chunked text labeled with part-of-speech and semantic types of noun phrases. With the help of base chunks, the system had some capability to skip adjuncts and prepositional phrases in pattern matching. Cohen et al. [29] is an ontology-driven, integrated concept analysis system. It extracted transport, interaction, and expression by using expanded (regular expression) patterns based on cell typing, syntactic trees, and anaphora resolution. Hakenberg et al. [47] automatically derived patterns from positive sentences, and aligned patterns against new text. It is noticeable that Hakenberg et al. [47] could generate new patterns by replacing the triggers in the pattern. But these patterns were defined on surface level and no syntactic analysis was employed.

In Chapter 4, we report a novel framework to facilitate the development of a pattern-based biomedical relation extraction system. We acknowledge several studies underlying our framework. The automated pattern generation employed in this study shares the fundamental assumptions of certain linguistic theories, such as Lexicalized Tree Adjoining Grammar (LTAG) [119], Head-Driven Phrase Structure Grammar [108], and Lexical Functional Grammar [13]. In particular, we believe that the concept underlying our method is similar to that of LTAG. The paradigm of inferring patterns exploited in our method shares the ideas with [24, 46, 66, 73], but we focus on a specific set of patterns pertaining to the expression of biomedical relations.

With regard to referential relations, numerous relationships between the concepts, for instance *is-a*, *member-of* and *part-of*, are discussed in general knowledge representation (e.g., KL-ONE) and biomdical concept ontologies (e.g., UMLS[1]). Some of these relations were considered in biomedical information extraction systems in order to improve their performance [87, 134, 135]. In this dissertation, we integrate them in our framework and examine their utility for biomedical relation extraction.

The studies of our framework leads to improve the representation of information in natural texts. We proposed Extended Dependency Graph (EDG) with the intuition to find a level of abstraction that is more suitable for tasks of relation extraction. We believe that the linguistic information and domain knowledge studied in the framework can be used in a more general way so that both pattern-based and machine-learning approaches of relation extraction can benefit.

2.2.2 Machine Learning-based Approaches

Relation extraction task can be treated as a classification problem [5].

$$f_R(T(s, e_1, e_2)) = \begin{cases} +1 & \text{if } e_1 \text{ and } e_2 \text{ are related according to relation } R. \\ -1 & \text{otherwise} \end{cases} \tag{2.1}$$

[1]UMLS® Reference Manual: http://www.ncbi.nlm.nih.gov/books/NBK9676/

In Equation (2.1), s is a sentence which includes e_1 and e_2 as two entity mentions, $T(s)$ is information that is extracted from s, and $f_R(.)$ decides whether the entities in the sentence are in a relation or not. As a classification problem, $f_R(.)$ can be constructed as a discriminative classifier. The classifier is then trained using a set of positive and negative relation instances.

Machine learning-based approaches are data-driven and can derive models for automated extraction from a set of annotated data [3, 8, 14, 65, 84, 114, 139]. The input to the classifier can be either a set of features extracted from sentences (feature-based methods) or rich structural representation like trees and graphs (kernel methods). Both methods can be used in discriminative classifiers including [7], support vector machines [137], conditional random fields [69] etc.

Feature-based methods represent labeled instances as a sequence of features. The problem of feature-based methods is that data cannot always be easily represented with explicit feature vectors. Since natural language processing applications involve structured representations of the input data, it can be difficult to select features which are good indicators of entity relations. To alleviate the problem of selecting subset features, specialized kernels are designed for relation extraction in order to exploit rich representation of the input data like parse trees and maximize performance.

Kernel-based methods attempt to solve this problem by implicitly calculating dot-products for every pair of examples. It is based on the "kernel trick" by replacing dot-products with some other choice of kernel [12]. Instead of extracting feature vectors from examples, they apply a similarity function between examples and use a discriminative method to label new examples. Compared with feature-based methods, kernel-based methods are easy to encode linguistic and semantic information, but have higher computation complexity.

Among the different machine learning-based approaches, SVMs are often used in conjunction with kernels [9, 14, 136]. For BioNLP relation extraction task, features can be extracted from non-linguistic information like co-occurrence word-pairs [3, 65] to rich linguistic information extracted from parse trees [8, 82] or their combination [26, 45, 84, 97, 113]. Tikk et al. performed a comprehensive benchmarking of nine different methods for protein-protein interaction extraction that use convolution kernels on rich linguistic

information [132]. Other samples of state-of-art biomedical relation extraction systems include [28, 80, 89, 117, 139].

Many kernel-based relation extraction systems have employed lexical and syntactic information [15, 96, 145]. There has been a growth in the use of more complex kernels and sophisticated parameter tuning methods to improve the results [25, 143]. In the protein-protein interaction task, machine learning methods using rich feature vectors [85], edit distance kernel [40], dependency tree kernel [27], all-path graph kernel [3], or their combination and variations [84, 144] have been proposed. Tikk et al. [132] summarized and compared these work on different corpora. However these methods lack the ability to consider an unified representation that can allow machine-learning methods to generalize more easily from textural variations.

In this thesis, our focus is on improving the representation of information in natural texts, rather than on developing new kernels. There have been several attempts to leverage syntax and shallow semantic argument structure [78, 86, 95, 98, 134, 135]. They offer insight on utility of information beyond syntax. We develop the EDG approach for relation extraction based on these ideas.

2.3 Biomedical Corpora

High-quality biomedical corpora are essential for the development of any type of relation extraction systems. Table 2.1 shows primary corpora so far. Some of them are used in our evaluations. We will describe them in more details in the related sections. A comparative analysis of corpora including AIMed [16], BioInfer [109], HPRD50 [42], IEPA [38], and corpus of LLL challenge [92] was described by [110] . Miwa et al. [88] also discuss one way to use multiple corpora.

Several shared tasks were organized as collaboration between teams to either exploit novel methods of information extraction, or develop system for realistic usage. The BioNLP Shared Task (BioNLP-ST) has been organized three times since 2009. The goal is to provide the community with shared resources for the development and evaluation of information

Table 2.1: Corpora for biomedical relation extraction.

Dataset	**Description**
AIMed	protein-protein interactions
BioCreAtIvE II and III	protein-protein interactions
BioCreAtIvE V	chemical-disease relations
BioInfer	protein-protein interactions
BioNLP ST GE corpora	GN, Phos, Trans, Loc, Binding, etc
BioText	disease-treatment relations
BLLIP Brown-GENIA treebank	hand-parses, no overlap with the GENIA treebank
Drug-Drug Interaction Extraction	drug-drug interactions
EDGAR	drugs, genes, and relations
GENIA	linguistic annotation, GENIA ontology, GENIA event, disease-gene association
HPRD50	protein-protein interactions
IE Data Sets	gene-disease relations, 856 sentence; protein-protein interactions, 5456 sentences
IEPA	protein-protein interactions
LLL	protein-protein interactions
PennBioIE	biomedical entity types and syntax
PICAD	protein-protein interactions
PICorpus	protein-protein interactions
The Anaphora Corpus	pronominal anaphora and their antecedents

extraction systems [62]. Kano et al. [55] propose a unified services to integrate nine existing individual event extraction systems.

Chapter 3

SENTENCE SIMPLIFICATION

Sentence simplification is a technique to detect various types of clauses and constructs contributing to the complexity of sentences, and to produce multiple simple sentences while maintaining the communicated message. Our hypothesis here is that by reducing the complexity, sentence simplification can ease the development of natural language processing and text mining tools.

This chapter describes types of simplification constructs that are frequently encountered in the biomedical literature. Moreover, we developed iSimp, a sentence simplification system, to reduce the syntactic complexity of a sentence [101, 102]. We show that our approach not only yields good performance but also can aid biomedical text mining and relation extraction applications, such as sentence ranking and selection, and open information extraction tasks in general. The web service and corpus can be found at http://research.bioinformatics.udel.edu/isimp.

3.1 Background

Many of text mining and information extraction tools detect the information in the text if it fits some common patterns reliably. For example, if the task is to detect phosphorylation information (e.g., <kinase, substrate>), we might look for sentences that are written in the form of "**X** *phosphorylates* **Y**", as shown in Exampe 1.

Example 1 It was suggested that **Yak1** *phosphorylates* **Crf1** to promote its nuclear entry.

This sentence mentions the phosphorylation in a format that is easy to process. However, we also see sentences containing complex grammatical structures. For example,

Example 2 Active **Raf-2** *phosphorylates* and activates **MEK1**, which in turn *phosphorylates* and activates the MAP kinases signal regulated kinases, **ERK1** and **ERK2**.

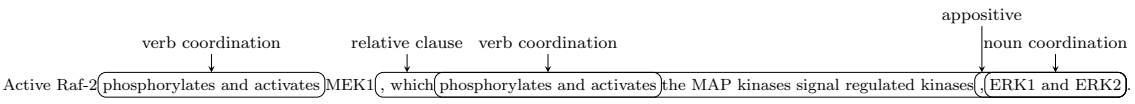


Figure 3.1: A sample sentence with simplification structures marked.

In Exampe 2, the <kinase, substrate> pairs are <Raf-2, MEK1>, <MEK1, ERK1>, and <MEK1, ERK2>.

Humans can easily grasp the phosphorylation information in this sentence and focus on the phosphorylation information alone. But an automated text mining system may not identify <MEK1, ERK2> as a <kinase, substrate> pair without complex rules and patterns. However, designing rules and patterns for all possible syntactic variations is impractical, because sentence constructions and writing styles vary considerably from one to another.

This chapter proposes an alternative approach to detect and extract information from complex sentences. Instead of matching all possible variations in text, we propose to simplify complex sentences first, and then attempt to match simple patterns to the simplified sentences. The hypothesis is that sentence simplification can alleviate the problems of text mining tools when dealing with complexities. For example, after identifying all constructs in Exampe 2 (see Figure 3.1), simplified sentences Examples 3a to 3c can be generated. It is now possible to extract the <kinase, substrate> pairs from Examples 3a to 3c, by using the simple pattern "**X** *phosphorylates* **Y**".

Example 3
a. Active **Raf-1** *phosphorylates* **MEK1**.
b. **MEK1** *phosphorylates* **ERK1**.
c. **MEK1** *phosphorylates* **ERK2**.

Based on this idea, we proposed to detect various constructs of a sentence to reduce its syntactic complexity. These simplification constructs include 11 types of coordinations, 2 types of relative clauses, appositions, subordinate clauses, 5 types of introductory phrases, and parenthesized elements.

To implement this idea, we developed iSimp, a sentence simplification system, to transform simplification constructs into BioC format that is easily accessible to text mining

tools [31, 101, 102]. Then we created a corpus marked with types of simplification constructs and evaluated the performance of iSimp on this corpus. The evaluation not only shows the promising results of sentence simplification, but also does so for the first time in terms of precision and recall in this domain. We also evaluated the impact of iSimp for the performance of various information extraction applications, including biomedical relation extraction task, sentence ranking and selection ($Rank_{\text{Pref}}$), and open information extraction tasks in general. These results lead us to use the sentence simplification in other frameworks described in this dissertation.

We made iSimp and this evaluation corpus publicly available. The corpus can then be used by other researchers for development and evaluation purposes.

3.2 Syntactic Simplification Constructs

We first define major syntactic constructs for simplification, then discuss the challenges that are frequently encountered during simplification. In the examples given throughout this chapter, the syntactic structures are enclosed in square brackets and words that **trigger** the potential presence of simplification are emphasized in bold.

Coordinations are complex syntactic structures that link together two or more conjuncts of syntactically equal status [52]. These conjuncts are linked by coordinating conjunctions, e.g., "and" in Exampe 4, "or", and "but". Sometimes correlative conjunctions are used together with coordinating conjunctions (e.g., "both ... and", "between ... and", "either ... or", "neither ... nor", etc.). These are helpful to mark the beginning of a coordination construct. Based on different categories of conjuncts, we further classify the coordination into 11 subcategories: noun/noun phrase, verb/verb group, preposition/prepositional phrase, adjective/adjective phrase, adverb/adverb phrase, and sentence coordination.

Example 4 An integral membrane protein can be found in $\big[$multivesicular bodies/lysosomes **and** secretory granules$\big]_{\text{noun phrase coordination}}$.

Usually, all conjuncts in a coordination belong to the same syntactic category and such category is determined by the head of the conjunct. In Exampe 4, the word "granules" is the head of "secretory granules", since it determines that the phrase is a noun phrase.

Table 3.1: Criteria for noun phrase similarity.

1. same word	6. uppercase words
2. numbers	7. containing hyphen/dash/slash
3. Greek alpha-beta	8. parenthesis elements
4. numbers followed by letters	9. common prefix
5. letters followed by numbers	10. common suffix

Table 3.2: Criteria for verb group similarity.

- They have same tense (e.g., present, past, future)
- They are of same grammatical number (singular or plural)
- They have same part-of-speech
- They are in same semantic group etc. (determined using DISCO [67])

iSimp uses the head words of phrases for the comparison. We assume the head word of a noun phrase is the last noun or pronoun, the head of a verb group is the first last (e.g., "changed" is the head of "had been changed"), and the head of a prepositional phrase is the preposition. We say two head words have the same type if they both follow the similarity rules listed in Tables 3.1 and 3.2.

Relative clauses are clauses that modify noun phrases [4]. The modified noun phrases are called *referred name phrases* and are underlined in our examples. Our method detects two categories of relative clauses. Full relative clauses are always introduced by relative pronouns, such as "which", "who", and "that" (Exampe 5a). Reduced relative clauses start with a gerund/past participle and have no overt subject (Exampe 5b). Although reduced relative clauses start with gerund or past participles, their presence does not always indicate the beginning of a construct. In Exampe 6a the mention of "limiting" does not mark the beginning of a reduced relative clause, while in Exampe 6b the mention does.

Example 5 a. This gene is composed of multiple exons, $\left[\textbf{which}\ \text{span at least five cosmids}\right]_{\text{full relative clause}}$.

b. A total of 11 additional families $\left[\textbf{carrying}\ \text{this mutation}\right]_{\text{reduced relative clause}}$ were identified.

Table 3.3: Criteria for apposition detection.

One of two noun phrases begins with a number, a determiner (e.g., “a”, “an”, “the”), or words “other” or “another”
• If one noun phrase contains a number or word “both”, the other one must contain a noun phrase coordination with the same number of conjuncts
• Both noun phrases can not be part of a noun or noun phrase coordination
• The first noun phrase can not be part of a introductory phrase

Example 6 a. the rate **limiting** enzyme in cholesterol synthesis

b. the enzyme $\left[\textbf{limiting}\text{ the endogenous pathway of cholesterol synthesis}\right]_{\text{reduced relative clause}}$

Appositions are constructs of two noun phrases next to each other, typically separated by a comma [19]. Both noun phrases refer to the same entity but one (appositive) serves to describe the other in a different way. For example, in the sentence below, the phrases “RB” and “the protein product of the retinoblastoma tumor-suppressor gene” are in apposition, with the appositive identified in the brackets.

Example 7 RB, $\left[\text{the protein product of the retinoblastoma tumor-suppressor gene}\right]_{\text{appositive}}$, regulates the activity of specific transcription factors.

An appositive generally occurs between two commas. However, not all structures that fall within two commas are appositives. For example, an “apposition” can simply be part of a noun phrase coordination. Although Examples 8a and 8b are textually similar, Exampe 8a needs a coordination, while Exampe 8b is an apposition.

Example 8 a. $\left[\text{eIF2alpha dephosphorylation, GADD34 and CreP}\right]_{\text{coordinatiion}}, \ldots$

b. Two markers, $\left[\text{D16S3070 and D16S3275}\right]_{\text{appositive}}, \ldots$

To decide whether a construct is an apposition, one of the criteria in Table 3.3 must be satisfied. Whenever we encounter a noun phrase followed by a comma, we look to see if there is a noun phrase after the comma beginning with a determiner (e.g., “D16S3275, *a* microsatellite marker”) or a number (e.g., “*two* markers, D16S3070 and D16S3275”), and so on. We further check if any or both of the noun phrases are outside coordination boundaries and introductory phrases.

Table 3.4: Examples of subordinate clauses, introductory phrases, and parenthesized elements.

Type	Example
Subordinate clause	Changes in expression of iNOS during late gestation have not yet been studied longitudinally in any species, $\lceil$**because** repeatedly taking biopsies could not be performed.$\rceil_{\text{subordinate clause}}$
Introductory phrase	$\lceil$To address this question,$\rceil_{\text{introductory clause}}$ we examined the transcriptional activation of the HIV-1 LTR, . . .
Parenthesized element	The CCN family of proteins is composed of six secreted proteins (CCN1-6), which are grouped together based on their structural similarity.

Table 3.4 shows examples of three other sentence simplification construct that iSimp can handle. **Subordinate clauses** (also known as dependent clauses) provide additional information to the main clause. They can stand alone as a sentence if the trigger word (subordination) is removed. In our approach, only subordinate clauses beginning with the subordinators are handled (e.g., "if", "although", "because", etc.). Clauses starting with "suggesting that", "providing that", etc. are also treated as subordinate clauses in our approach.

Introductory phrases are phrases that open sentences other than the usual subject-verb order. Our approach handles five types of introductory phrases: gerund-participle phrases (e.g., "utilizing Xenopus laevis as our model"), past-participle phrases (e.g., "given this"), to-infinitival phrases (e.g., "to do this"), prepositional phrases (e.g., "in the present study"), and adverbs (e.g., "recently").

Parenthesized elements refer to any words enclosed within "()", "[]", and "{}". They usually refer to or describe preceding noun phrases. Note that reference numbers, section numbers, and itemized lists might also be enclosed within these brackets.

3.3 Generation of Simplified Sentences

After detecting the simplification constructs and their referred noun phrases, the simplifier generates separate sentences for each. For a coordination, the original sentence can

be split into multiple ones, each containing one conjunct. For instance, Exampe 4 on Page 16 can be simplified as in Examples 9a and 9b.

Example 9 a. An integral membrane protein can be found in $\left[\text{multi-vesicular bodies/lysosomes}\right]_{\text{conj}}$.

b. An integral membrane protein can be found in $\left[\text{secretory granules}\right]_{\text{conj}}$.

We note here that splitting noun phrase conjunctions plays an important role in information extraction tasks. Although the simplified sentences may not necessarily appear to be simpler, the patterns for information extraction would be more likely to be applicable to these sentences.

A sentence containing a relative clause can be simplified into two sentences: one that skips over the relative clause and the other that combines the referred noun phrase with the relative clause. For instance, Exampe 5b on Page 17 can be simplified as in Examples 10a and 10b.

Example 10 a. A total of 11 additional families were identified.

b. 11 additional families **are** $\left[\text{carrying this mutation}\right]_{\text{reduced relative clause}}$.

For appositions and parenthesized elements, the original sentence can be split into two, one containing the referred noun phrase, the other containing the appositive (or parenthesized elements). For instance, Exampe 7 on Page 18 can be simplified as in Examples 11a and 11b.

Example 11 a. RB regulates the activity of specific transcription factors.

b. $\left[\text{The protein product of the retinoblastoma tumor-suppressor gene}\right]_{\text{appositive}}$ regulates the activity of specific transcription factors.

Depending on the application, introductory phrases are removed or placed at the end of the sentence. Subordinate clauses can be separated as independent sentences by removing the subordinations.

3.4 iSimp

To make sentence simplification interoperable with other natural language processing and text mining (NLP/TM) applications, we see a sentence simplifier as a module to be used at the beginning of NLP/TM pipelines. With this in mind, we designed and developed iSimp. Figure 3.2 shows how iSimp can be used as a module in an NLP/TM pipeline. It is

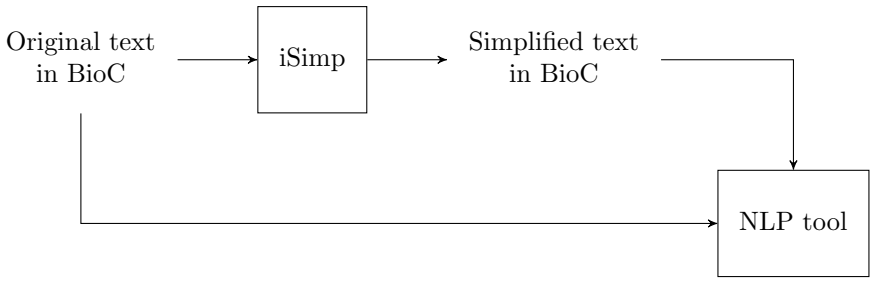


Figure 3.2: NLP pipeline with iSimp.

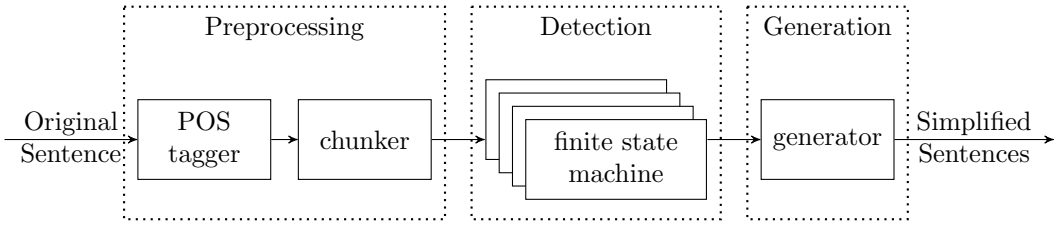


Figure 3.3: The architecture of the sentence simplification system.

worth pointing out that iSimp is designed to act like an optional plug-in. This means other applications are not expected to make changes to use iSimp. Instead, we should be able to plug iSimp in/out as needed, where the application can access original sentences, simplified sentences or both.

iSimp contains three stages: preprocessing the original sentence, detecting simplification constructs, and generating simplified sentences (see Figure 3.3).

Preprocessing module takes a raw sentence as the input. Given the sentence, we first apply a part-of-speech tagger to determine the corresponding linguistic category of each word in the sentence. These categories can be nouns, verbs, prepositions, etc. We trained a tagger using maximum entropy model [7] on the GENIA corpus [130]. Next, we applied a shallow parser to group together words into noun phrases (NP), verb groups (VG), prepositional phrases (PP), and others (O). An example of a chunked sentence is given in Figure 3.4. Our shallow parser was also developed and trained using maximum entropy on the GENIA corpus.

Recursive transition network is used to detect simplification constructs. For each type of simplification, call it C, we construct a finite-state recursive transition network, M_C. Three outcomes are possible for each machine: (1) **success**, where the machine reaches a

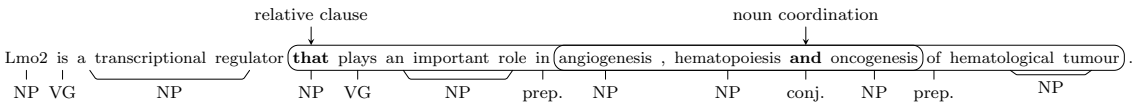


Figure 3.4: A sample sentence with chunks and simplification structures marked.

final state indicating the simplification structure is detected; (2) **failure**, where the machine reaches a final state denoting the structure requirements are not met; and (3) **pending**, when it is conjectured that another structure is nested inside and should be detected first.

The pseudocode for the detection algorithm is provided in Algorithm 1. Generally, the input sentence is scanned from left to right. Whenever a trigger word is encountered marking the presence of one of the simplification types, we call the corresponding finite state machine. Two lists, "complete" and "pending", are used to store instances of these machines. If the instance finishes in a success state, then we put it in the "complete" list. Otherwise, we put it in the "pending" list, as this indicates potentially nested constructs.

Algorithm 1 Algorithm for simplification detection.

```
1:  procedure DETECT_SIMPLIFICATION
2:      for each words w of a sentence do
3:          if w is a trigger word of construct C then
4:              start and run an instance of machine M_C, say m_C
5:              if m_C is pending then
6:                  push m_C into pending list
7:              else
8:                  push m_C into complete list
9:              end if
10:         end if
11:     end for
12:     if pending list is not empty then
13:         solve nested cases using easy-case-first strategy
14:     end if
15: end procedure
```

When two or more constructs are nested, we apply the easy-case-first strategy. Let us consider the relative positions of two constructs, C_1 and C_2. There are two cases which might confuse the finite state machines M_{C_1} and M_{C_2}: C_2 is nested inside of C_1, or C_2 is on the right of C_1. Consider the sentence in Figure 3.4 for an example of the former case. The first

trigger in this sentence is "that", which marks the presence of a relative clause. However, the finite state machine returns a "pending" outcome for the relative clause when it encounters the trigger word "and". Thus, the coordination is attempted first. Upon its successful detection, the simplifier returns to the detection of the relative clause, which is successfully done, by skipping over the coordination previously marked.

Left boundary detection of a coordination is particularly challenging, as it is not always clear how many conjuncts are involved. Commas are not always helpful in identifying the exact location of the left boundary. Sometimes, a coordination can be part of an appositive (as we saw in Exampe 8a on Page 18), and sometimes a coordination can follow immediately after an introductory phrase, both of which are also marked by commas.

Even in cases where there are only two conjuncts, there may be multiple candidates to consider. Consider Exampe 12, for instance. Here, we might incorrectly mark "the IGF-I promoter" and "an ApaI polymorphism" as the two conjuncts. In addition to part-of-speech information, the similarity of conjuncts can help in some cases. In Exampe 12, both conjuncts should have an "NP of NP" sequence.

Example 12 Glucose-stimulated insulin secretion uses hyperglycemic clamps in carriers of $\big[$a CA repeat in the IGF-I promoter **and** an ApaI polymorphism in the IGF-II gene$\big]_{\text{noun phrase coordination}}$.

Right boundary detection of simplification constructs is also challenging. Even for a simple coordination, such as "B and T cells", we need to recognize that "cells" refers to both "B" and "T", and thus mark the coordination as "$\big[$B **and** T$\big]$ cells". For nested cases where we have one construct followed by another, we need to determine which construct should be identified first. The following sentence shows three levels of relative clause nesting.

Example 13 We identified a chromosome translocation $\big[$associated with ADPKD $\big[$that disrupts PBP $\big[$encoding a 14 kb transcript in the PKD1 candidate region$\big]_{\text{red. rel. clause}}\big]_{\text{full rel. clause}}\big]_{\text{red. rel. clause}}$.

Referred noun phrases detection is important to identify for the generation of simplified sentences. The challenge here is when we have a sequence of nouns and prepositions. In the sentence "A total of 11 additional families $\big[$**carrying** this mutation$\big]_{\text{reduced relative clause}}$

were identified.", for instance, it is not clear whether "A total of 11 additional families" or "11 additional families" is the antecedent for the relative clause.

3.5 Evaluation

We conducted two types of evaluation. First, we tested the performance of iSimp. Then we evaluated whether iSimp can aid in improving sentence selection and open information extraction applications.

3.5.1 Evaluation of Sentence Simplification

We evaluated our simplifier for 6 types of simplification constructs: coordinations, relative clauses, appositions, subordinate clauses, introductory phrases, and parenthesized elements. Such evaluation is non-trivial and has no precedence, as previous works focused only on the impact of sentence simplification. We evaluated our simplifier for the detection of syntactic constructs using manually annotated corpus.

Our simplifier was constructed based on a development data set consisting of MEDLINE abstracts concerning proteins and genes. For evaluation, we randomly selected 100 MEDLINE abstracts (a total of 998 sentences), having the words "protein" and "gene" in the title. We asked five judges to mark the simplification constructs. To provide a high quality annotated corpus, each sentence was annotated by two judges independently and annotation conflicts (57 sentences in total) were resolved by a third party opinion.

As shown in Table 3.5, there are 526 coordination instances annotated in the corpus. Out of 998 sentences in 100 abstracts, 53% sentences contain at least one coordination, and almost every abstract (98%) contains coordination. Together with statistics of other simplification constructs in Table 3.5, we observe that in the abstracts of scientific articles, authors intend to use long and complicated sentences to summarize, in few sentences, the various facts described throughout the manuscript.

Here we report two sets of results: (1) results on the assignment of the simplification type to a construct; and (2) results on the detection of the construct boundaries. Consider the following sentence as an example.

Table 3.5: Statistics of PubMed corpus for sentence simplification detection.

Types	Instances	Sentences	Abstracts
Coordination	526	53 %	98 %
Relative clause	244	24	85
Apposition	43	4	31
Subordinate clauses	31	3	26
Introductory phrases	147	15	71
Parenthesized elements	287	29	84
Total	**1,278**	**–**	**–**

Example 14 An integral membrane protein can be found in $\lceil$multivesicular bodies/lysosomes **and** secretory granules$\rceil_{\text{noun phrase coordination}}$.

In the first evaluation, if we were able to detect a noun phrase coordination in the sentence, we counted it as one true positive. Using this measurement, we report an average precision of 99.6%, recall of 90.5%, and F-score of 94.8% (Table 3.6). The majority of false negatives in the first evaluation were in the case of coordinations. In our approach, we do not rely only on lexical and syntactic information. We also look at the similarity of the conjuncts to be considered in a coordination. Although including the similarity feature helps identify the proper coordinations more often than it hurts, some errors are inevitable. Cases like "we measured [m(b), and food intake] ..." are missed by our simplifier. In the future, rather than not attempting any coordination detection when there is no similarity, we may default to selecting the closest two candidates. Most of the remaining false negatives were attributed to reduced relative clauses. Missed cases were due to the overreliance on the part-of-speech tagger. Since one of the trigger words for a reduced relative clause is a past participle verb, we ignored the cases in which the verb was erroneously tagged as a past tense verb. To address this issue, we plan to use contextual clues besides part-of-speech.

For the second evaluation, we need to detect both the simplification type and the boundaries of the construct. So in the above example, we counted it as one true positive only if we were able to correctly find the coordination starts with "multivesicular" and ends with "granules". Using this measurement, we report an average precision of 90.9%, recall of 91.9%, and F-score of 91.4%. While there is hardly any drop in the recall, we note that

Table 3.6: Results for simplification detection. Performance is reported in terms of Precision, Recall, and F-score.

Types	Assignment			Assignment+Boundary		
	P	**R**	**F**	**P**	**R**	**F**
Coordination	100.0	87.9	91.7	76.8	85.5	80.9
Relative clause	100.0	93.0	96.4	88.5	91.3	89.8
Apposition	93.8	83.3	88.2	93.8	83.3	88.2
Subordinate clause	100.0	96.8	98.4	100.0	96.8	98.4
Introductory phrase	100.0	95.9	97.9	97.1	95.8	96.5
Parenthesized element	100.0	95.9	97.9	100.0	95.9	97.9
Average	**99.6**	**92.4**	**95.9**	**90.9**	**91.9**	**91.4**

this time the issue is with false positives relevant to the boundary detection. In the case of coordinations, half of false positives were attributed to erroneously attaching a left noun to the first conjunct or a right noun to the last conjunct rather than to the entire coordination (Exampe 15a). In the case of relative clause detections, a majority of false positives were due to nesting constructs, involving a coordination, relative clause, or appositive (Exampe 15b).

Example 15 a. We further investigated the occurrence and frequency of $_{\text{wrong}}\big[$gene $_{\text{correct}}\big[$amplification and over-expression$\big]\big]$ affecting RHBDD2 in 131 breast samples.

b. The results $\big[\big[$presented here$\big]_{\text{correct}}$, and those of previous studies$\big]_{\text{wrong}}$, suggest that . . .

3.5.2 Evaluation on Phosphorylation Extraction

To demonstrate the impact of sentence simplification on applications, we first evaluated the utility of sentence simplification for rule-based systems. Our hypothesis is that with sentence simplification, the number of extraction patterns could be greatly reduced, while maintaining or even improving the system performance. Fewer extraction patterns also mean an advantage in terms of system maintainability, runtime, scalability, as well as portability to the extraction of other post-translational modification types.

For this purpose, we evaluated the utility of iSimp in RLIMS-P, which a rule-based system for protein phosphorylation extraction [50, 91]. We retrieved 1,000 MEDLINE

abstracts containing trigger words ("phosphorylat" + "e", "es", "ted", and "ion"). 2,010 sentences were identified to contain both a trigger word(s) and a protein mention(s), based on the outputs of the RLIMS-P pre-processing modules. In these sentences, 2,824 pairs of <trigger, protein> were detected. Of them, 1,768 were identified as trigger-argument pairs by high-precision patterns. We took these to be valid pairs after a quick manual review. After applying sentence simplification, we extracted 343 additional pairs, which were manually inspected and confirmed as valid pairs. Overall, simplification allowed us to correctly extract phosphorylation information from nearly 20% more sentences using RLIMS-P rules and patterns.

3.5.3 Evaluation on Sentence Selection

We also evaluated the utility of sentence simplification for machine-learning systems. Here we chose $\text{Rank}_{\text{Pref}}$ which deals with the ranking and selection of sentences containing a given gene name, a given relevant term, and the description of a relationship between the two [133].

The ranking and selection of sentences are based on a model that is trained on annotated data. One important feature used in the learning process marks the presence or absence of a syntactic relationship between the gene and the relevant term. This relationship is determined based on pattern matching applied at the syntax level of the sentence. The hypothesis here is that matching of the patterns can be improved if sentence simplification is used.

Two evaluations were conducted. The first evaluation reports the performance of the learned model to pick one sentence from a pair of sentences. When sentence simplification was applied, we observed an increase of 5.5% from 74.42% to 79.90%, which represents a relative increase of 7.4% in the tool's performance. The second evaluation reports on the performance of the learned model to choose one sentence from a group of sentences. We observed an increase of 7% from 67% to 74%, which represents a relative increase of 10.4% in the tool's performance when the sentence simplifier was incorporated.

3.5.4 Evaluation on Open Information Extraction

Information extraction tools focus on a specific task, which tends to be precise and narrow. Open information extraction (OIE) has been introduced as a new paradigm, where a system extracts a large set of relational tuples without requiring human input [41, 53]. For the extraction of a large set of relational tuples, one has to design syntactic patterns that can be matched in sentences, regardless of the domain from where they were obtained.

The patterns, used in the previous section for detecting syntactic relations, are general enough for the detection of relational tuples no matter the domain. We wanted to see how well these tuples are detected in sentences, with and without simplification, as this can give us an idea of the impact of simplification on OIE systems.

Out of 811 sentences annotated with a gene and a relevant term, 376 sentences matched the syntactic patterns in their original format. After the simplifier was applied, an additional 103 sentences matched the syntactic patterns via a simplified sentence. This translates into a 51.33% relative increase in recall. Note here that the remaining 242 sentences in which no pattern was matched are in majority sentences for which no relationship could be determined between the two annotated terms.

3.6 Sentence Simplification in BioC Format

BioC is a framework that aims to provide an easy and powerful way of integrating text mining tools [31]. It uses an XML format, which enables the sharing of documents and annotations (e.g., part-of-speech tags, named entities, and entity relations). Many NLP tools incorporated the BioC format. They perform tasks such as abbreviations, semantic role labeling, and gene normalization.

The integration of iSimp with the BioC format is somewhat different from those cases because iSimp produces new sentences besides the tagging of the original text. In addition, iSimp will sometimes introduce new words in the simplified sentence to keep it grammatically correct. For example, we put “is a” between the appositive clause and the singular noun phrase it refers to, to form a new sentence. Therefore, adding new words to the corpus is one of the factors that distinguish iSimp from other applications that enhance BioC.

To address this challenge, we designed and proposed a new schema of using BioC framework, which was not directly addressed in the original design of BioC [31]. The new schema can include words that are not part of the original text. Thus other than the text simplification, it can also be used in tasks of transliteration, query expansion, or document summarization.

For the task of sentence simplification, we first define a BioC tag set for annotating and sharing the simplification results [104, 106]. Figures 3.5 and 3.6 show the key file used in iSimp to define the semantics associated with the data. We use the annotation element to mark up the simplification construct components, and we use the relation element to specify how these components are related. In the latter, we further specify the name of the simplification type (e.g., coordination, relative clause, etc.), as well as roles for each component in the relation using the node element (e.g., "conjunct" and "conjunction" for the coordination, "referred noun phrase" and "appositive" for the apposition). For example, Figure 3.7 shows the coordination "phosphorylates and activates" in BioC format. This coordination contains two conjuncts ("phosphorylates" and "activates") and one conjunction ("and"). Some attributes, like the location elements, are not shown in this figure for lack of space.

As mentioned before, iSimp generates new simplified sentences. In iSimp, we include both original and simplified sentences in the BioC file. The offsets of the original sentences are the same as in the original text. However, the offsets of the simplified sentences start with the offset of the next character after the last character in the original document (offset+length of the document). This new collection could then be treated as the input collection for the next step in the NLP pipeline. Figure 3.8 shows an example of simplified sentences in the BioC format (left), as well as the corresponding text file (right) with locations highlighted.

To link text in simplified sentences to that in the original sentence, we introduce the **equ** (equivalence) relation. Figure 3.9 shows an example of an equivalence relation, in which we link "phosphorylates" back to the original sentence. This way phrases in the simplified sentences can be mapped back to the corresponding phrases in the original sentence. Equivalence relations can be used to ensure that downstream applications recognize the duplicated nature of such **equivalent** phrases and do not report the same information multiple

```
This key file defines the simplification constructs in the BioC XML file.

collection: This collection is an abstract from PubMed article.
  source: PubMed
  date: yyyymmdd. Date this example was created.
  document: this collection contains one document.
    id: PubMed Identifier (PMID)
    passage: the second sentence from the abstract
      infon type: abstract
      offset: abstract arbitrarily starts at 0.
      sentence: one sentence of the passage as determined by the opennlp
                sentence splitter.
        offset: a document offset to where the sentence begins in the
                passage. The sum of the passage offset and the local offset
                within the passage.
        annotation:
          infon type: simplification construct.
          location: location of the annotated text.
          text: the annotated text
        relation: there are 3 types of simplification constructions: coordination,
                  relative clause, and apposition. Each described
                  separately below
          coordination:
            infon type: coordination
            node: conjunct (there should be 2 or more conjuncts) and conjunction
          relative clause:
            infon type: relative clause
            node: referred noun phrase and relative clause
          apposition:
            infon type: apposition
            node: referred noun phrase and appositive
          parenthesis:
            infon type: parenthesis
            node: referred noun phrase and parenthesized elements
```

Figure 3.5: The key file used in iSimp to define the simplification constructs associated with the data.

```
This key file defines the simplified sentences in the BioC XML file.

collection: This collection is an abstract from PubMed article (PMID-8557975).
  source: PubMed
  date: yyyymmdd. Date this example was created.
  document: this collection contains one document.
    id: PubMed Identifier (PMID)
    passage: the second sentence from the abstract
      infon type: abstract
      offset: abstract arbitrarily starts at 0.
      sentence: the first sentence is the original sentence. The following
                sentences are simplified sentences.
        infon type: original sentence or simplified sentence
        offset: the original sentence have the same offsets. The simplified
                sentences' offsets start with passage.offset + passage.length.
        text: the original UTF-8 Unicode text as it appears in the original
              document.
        annotation: tokens in the sentence
          infon type: token
          location: location of the annotated text.
          text: the annotated text
      relation: map tokens in the simplified sentences to tokens in the
                original sentence.
        infon type: equ
        node role: original. Token in the original sentence
        node role: simplified. token in the simplified sentences. There might be
                   several "node simplified", if one token in the original
                   sentence appears several times in the simplified sentences.
```

Figure 3.6: The key file used in iSimp to define the simplified sentences associated with the data.

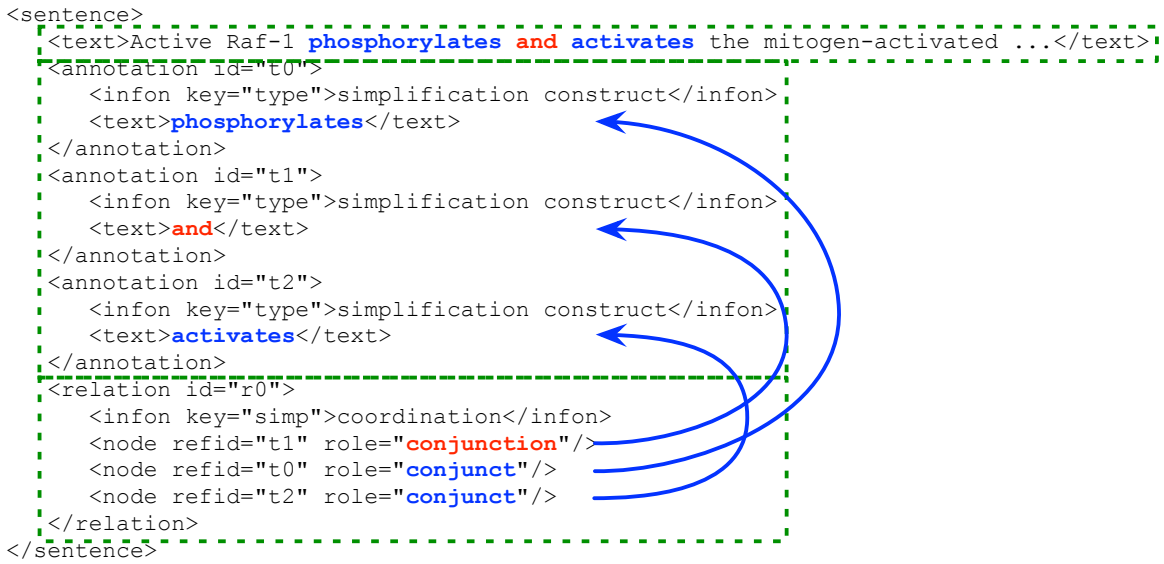


Figure 3.7: An example of sentence simplification annotation in BioC format. The coordination contains two conjuncts ("phosphorylates", "activates") and one conjunction ("and"). Some attributes, like the location elements, are not shown for the sake of space.

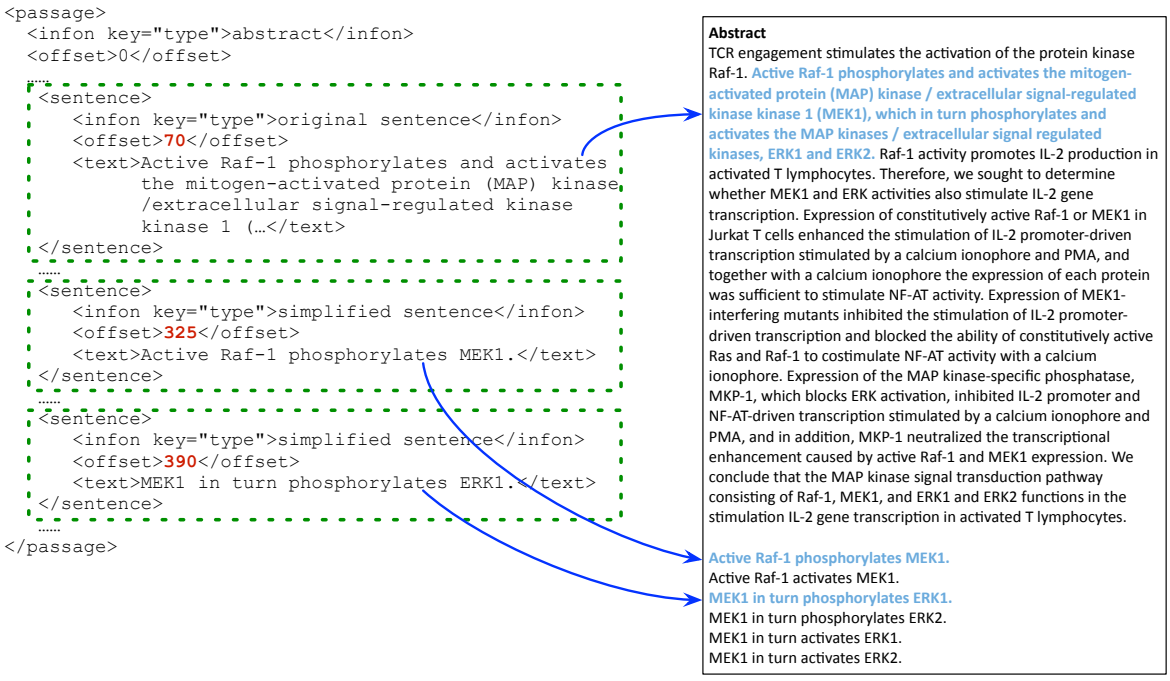


Figure 3.8: An example of simplified sentences in BioC format (left) and the corresponding text file (right) with locations highlighted.

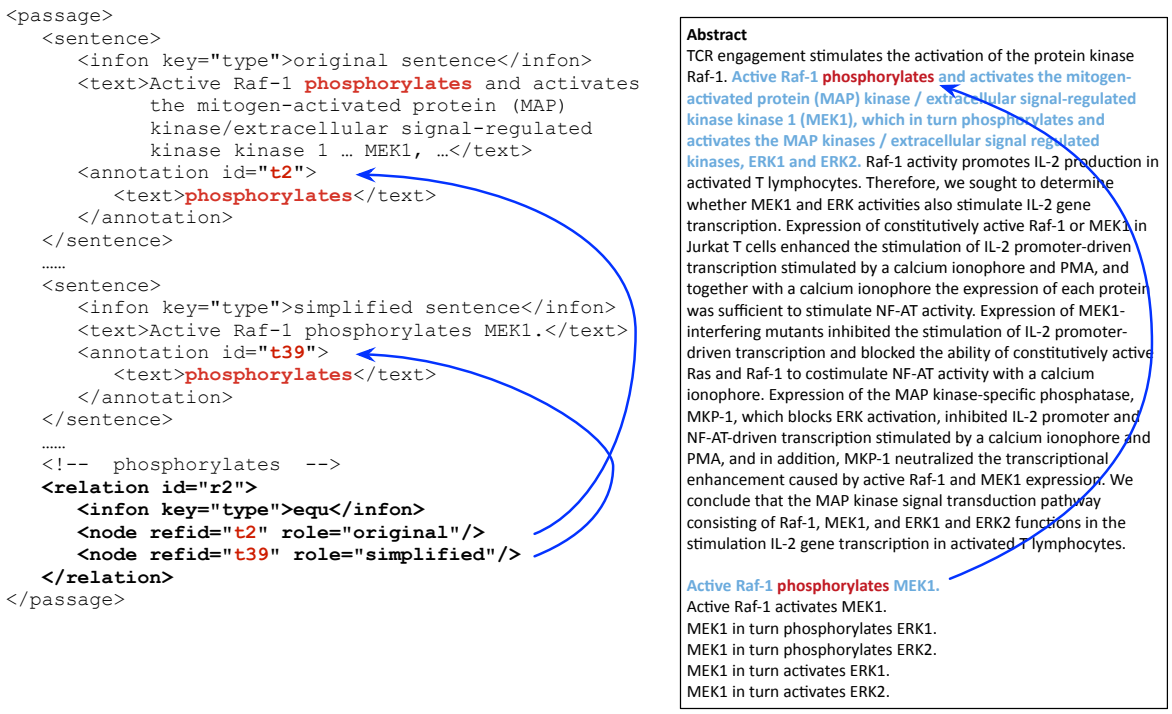


Figure 3.9: An example showing "equ" (equivalence) relations in iSimp-generated BioC file.

times in the end. Implementation of this mechanism was feasible owing to the extensibility of the BioC format.

3.7 Conclusions

In this chapter, we demonstrate that sentence simplification is important and useful for existing text mining applications that rely on the analysis of sentence structures. We describe our approach to detect various simplification constructs commonly found in the biomedical text and how to simplify complex sentences into simpler ones. With this in mind, we have developed iSimp, a sentence simplification system. By using BioC as the data-sharing communication medium, iSimp can be plugged in the BioC-compatible framework to collaborate with other applications for building an integrated biomedical text mining system. We evaluated the performance of iSimp and showed that the use of iSimp can improve the results of different information extraction tools.

Chapter 4

A FRAMEWORK FOR PATTERN-BASED BIOMEDICAL RELATION EXTRACTION SYSTEMS

A relation extraction system achieving high performance is expensive to develop because of the substantial time and effort required for its design and implementation. This chapter reports a novel framework to facilitate the development of a pattern-based biomedical relation extraction system. It has several unique design features: (1) leveraging linguistic knowledge about syntactic variations possible in a language and automatically generating extraction patterns in a systematic manner, (2) applying sentence simplification to improve the coverage of extraction patterns, and (3) identifying referential relations between a syntactic argument of a predicate and the actual target expected in the relation extraction task.

Two relation extraction systems derived using the proposed framework are discussed. (1) A Genia relation extraction system is evaluated on two data sets of the BioNLP-ST 2011 GE corpus and a comparative analysis of various implementations of our framework was conducted to evaluate the importance of each system module. (2) An miRNA-gene relation extraction system is evaluated on the in-house corpus and Bagewadi et al. corpus. Both analyses indicate that without increasing the number of patterns, simplification and referential relation linking play a key role in the effective extraction of biomedical relations.

4.1 Background

With the rapid growth of biomedical publications, how a scientist can be helped to keep up with the new findings reported in the literature? A popular idea is to turn unstructured text into structured by extracting semantic information. However the high volume of literature makes human annotation impossible. As a consequence, we have observed an increase in the

effort spent on automatically extracting information from research literature and developing biomedical text mining tools.

Relations between entities is the most fundamental structure of our interest. To extract relations from the literature, pattern-based systems are well studied for decades because they can manipulate domain-specific knowledge to extract relations in a precise way. Moreover, they do not require annotated data to train a system. However, pattern-based approaches require domain experts to be closely involved in the design and implementation of the system to capture the patterns used for extracting the necessary information. In all cases, rigid extraction patterns are manually encoded in the systems. Owing to rigid patterns, pattern-based approaches usually achieve a high precision but are often cited for low recall. While it is feasible to manually identify and implement high-quality patterns to achieve a good precision, it is often impractical to exhaustively encode all the patterns necessary for a high recall in this manner. The work reported in this chapter enables the fast development of pattern-based systems, while mitigating some of these concerns. We aim to reduce the involvement of domain experts and their manual annotation, and to attain high precision and recall.

Our approach starts by identifying a list of triggers for the target relation (e.g., "associate" for the binding relation) and their corresponding Trigger specifications (e.g., the number and type of arguments expected for each trigger). Given this information, we make use of linguistic principles to generate several lexico-syntactic patterns in a systematic manner. These lexico-syntactic patterns are matched with the input text in order to extract target relations.

To improve the applicability of the generated patterns, we incorporate two additional design features. The first is the use of text simplification. This allows us to design a small set of lexico-syntactic patterns to match simple sentence constructs, rather than try to account for all complex syntactic constructs by generating an exhaustively large amount of patterns. Second, the framework exploits referential relations. With this method, two phrases referring to the same entity (e.g., coreference relation) or in a particular relation (e.g., meronymy relation, also known as part-of relation) are detected in the text, and links are established between them. These links can be used when seeking the most appropriate phrase referring

to the target entity and, hence, allow for extraction of target entities beyond lexico-syntactic patterns.

The proposed approach is based on the mathematical formalism used for the description of natural language syntax, including those at the interface to semantics, rather than task-specific knowledge. Therefore, it is generalizable for different trigger words and potentially applicable to many different types of information targeted in biomedical relation extraction tasks. Our framework makes it possible to quickly develop relation extraction systems for different biomedical tasks, and requires only little input from a domain expert.

To evaluate the framework, we test it by producing two extraction systems. One for six relations that were part of the BioNLP-ST 2011 GE task and the other for miRNA-gene, miRNA-target, and gene-miRNA regulation relations. We show that by just taking the specification of trigger words (root word only), we produce two relation extraction systems with results that compare favorably with state-of-art results. We further show that we can achieve good precision and recall with the patterns generated from the trigger, and that simplification and referential relation linking can increase the recall without compromising the precision.

4.2 Methods

4.2.1 Architecture Overview

The architecture of our framework has several components (Figure 4.1), as summarized below and detailed in Sections 4.2.2 to 4.2.6.

In the literature, authors often use some **words** to alert the readers a potential occurrence of a relation. For example, the word "phosphorylates" in "JNK phosphorylates NFAT4 on two sites" is a sign of a "phosphorylation relation" and the word "transcribed" in "the FasL gene is transcribed" is a sign of a "transcription relation". Although the appearance of these words does not necessarily indicate that a relation can be extracted, rule-based systems still use them to initiate a matching process. And we call these words "triggers" in our framework.

More specifically, our framework requires **Trigger specifications** as inputs to locate the relations of interest. The framework consists of four system modules (Pattern generation,

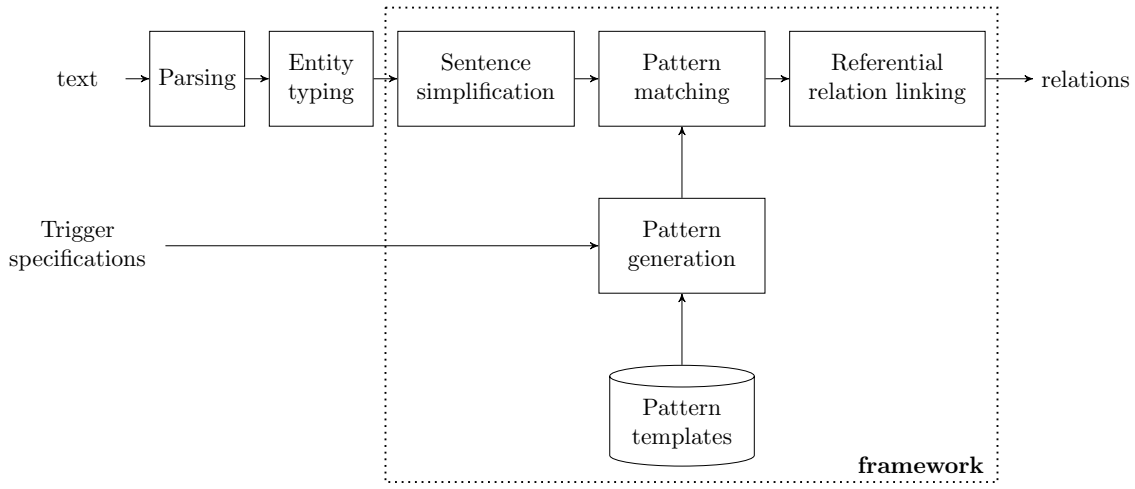


Figure 4.1: The framework of pattern-based biomedical relation extraction systems.

Pattern matching, Sentence simplification, and Referential relation linking) and two external modules (Parsing and Entity typing).

An example trigger specification is shown below:

Trigger 1 *phosphorylate* (1)
$\langle type \rangle = phosphorylation$ (2)
$\langle frame \rangle = Frame{:}NP_0/NP_1$ (3)
$\langle NP_0\ type \rangle = protein$ (4)
$\langle NP_1\ type \rangle = protein$ (5)
$\langle NP_0\ role \rangle \leftarrow agent$ (6)
$\langle NP_1\ role \rangle \leftarrow theme$ (7)

In the above example, Line (1) shows the trigger word, "phosphorylate" in this case. Line (2) indicates that it is the trigger for the relation "phosphorylation". Line (3) specifies that the trigger syntactically chooses two noun phrases, designated as NP_0 and NP_1. For further information, please refer to the later subsections on frames. Lines (4)–(5) add selectional restrictions, by requiring NP_0 and NP_1 to be proteins. Lines (6)–(7) show that if NP_0 and NP_1 can be extracted, and if both NP_0 and NP_1 meet the above constraints, then the framework will assign their semantic roles of agent and theme, respectively.

Now consider the example sentence "**The c-Jun amino-terminal kinase** *phosphorylates* **NFAT4**." From the sentence, we will extract "the c-Jun amino-terminal kinase" as the agent and "NFAT4" as the theme of the phosphorylation relation. This extraction is the

result of matching the text fragment with a pattern that is partly derived from the trigger specification. This pattern should not only capture the general syntactic form of a clause involving a transitive verb in an active voice, but also capture the selection restrictions imposed by the word "phosphorylates" and the arguments. Thus, this pattern contains the information described in two places: (1) lexical trigger that specifies the arguments, the selection restrictions on the argument, and the role they play, and (2) the syntactic constraints corresponding to different constructs (in this example, the active clause). We call the former **Trigger specification**, and the latter **Pattern templates**. Actual lexico-syntactic patterns are obtained by merging the trigger specifications and pattern templates. As we shall see later (Section 4.2.2), the combination of these two is mediated by the frame that is mentioned in the trigger specification.

We now briefly discuss four modules of the system architecture: Pattern generation, Pattern matching, Sentence simplification, and Referential relation linking (Figure 4.1).

The **Pattern generation** (Section 4.2.4) module uses trigger specifications and predefined pattern templates to derive lexico-syntactic patterns for each trigger word. The **Pattern matching** (Section 4.2.4) module then matches fragments of text with lexico-syntactic patterns, and extracts the textual expressions in the trigger and argument positions. In order to more effectively match with the patterns, the **Sentence simplification** (Section 4.2.5) module is used to preprocess the input text. It aims to ensure that the lexico-syntactic patterns generated in the previous step are able to be matched even in complex sentences. Finally, the **Referential relation linking** (Section 4.2.6) module links arguments identified by the pattern matching module with the target entities they refer to, where applicable. This step enables the system to find relations between more appropriate entities than the ones referred by textual expressions in the argument position.

As mentioned above, the framework needs the named entity mentions (e.g., gene or protein) for pattern selection restrictions and the syntactic structures for pattern matching. To obtain the two types of information, we employ two more modules. One is the *Entity typing* module, which assigns semantic types or categories to noun phrases. We have found that their use of semantic types enhances the precision of relation extraction [50, 90]. The other is the

Parsing module, which is used by both sentence simplification and pattern matching steps.

4.2.2 Trigger Specification

Trigger specifications are used to locate triggers and arguments in text for target relations. They are user-defined and are the only things that a user needs to provide. To make it easier to specify triggers, we ask users to provide the trigger's root, which is the primary lexical unit of a word. From the root morpheme, we can derive other forms of triggers using our previous work [83]. For example, from "phosphorylate" we derive "phosphorylates", "phosphorylated", "phosphorylation", etc. Automatically derived words may not be correct and we request users to select correct forms.

Next, we show two example trigger specifications for the same root morpheme, "express", but with different semantic types, gene and RNA, for the argument.

Trigger 2 *express.01*
$\langle relation \rangle = Gene\_expression$
$\langle frame \rangle = Frame{:}NP_0/NP_1$
$\langle NP_1\ type \rangle = gene$
$\langle NP_1\ role \rangle \leftarrow theme$

Trigger 3 *express.02*
$\langle relation \rangle = Transcription$
$\langle frame \rangle = Frame{:}NP_0/NP_1$
$\langle NP_1\ type \rangle = RNA$
$\langle NP_1\ role \rangle \leftarrow theme$

Although they share the same trigger, the two relations are of different types. The *Gene_expression* relation requires its theme (NP_1) to be a gene, and the transcription relation requires its theme (NP_1) to be of type RNA. Argument types in the trigger specification are essential to the framework because they emphasize the selection restrictions on arguments, and thus aid in achieving a high precision.

Relations can be categorized as **directional** or **non-directional**. For example, *Phosphorylation* relation is directional but Binding relation is non-directional. This is because "A phosphorylates B" and "B phosphorylates A" represent two different relations, but "A binds B" and "B binds A" are used to specify the same relation between "A" and "B". As a consequence, triggers associated with relations can be categorized as directional or non-directional as well

(e.g., "phosphorylate" for the Phosphorylation relation vs "bind" and "associate" for the Binding relation).

For the non-directional trigger, the two agents/themes undergo the same effect described by the trigger. Therefore, syntactically, either of the two arguments can be aligned with the subject (their order can be swapped). To distinguish non-directional triggers from the directional ones, we use an additional boolean constraint "⟨direction⟩= directional/non-directional".

4.2.3 Pattern Templates

A pattern template is specified by a sequence of words/phrases $\beta_1, \ldots, \beta_n$, followed by a set of constraints. Each constraint assigns a value for one of the β_i attributes.

In general, the pattern template specifies the predicates and arguments. To reduce the number of pattern templates, we limit pattern templates to capture one argument at a time. So the pattern templates will capture pairs <trigger, NP_i>. After templates are instantiated and arguments are extracted, we combine pairs if they have the same trigger. Thus, we can extract relations with multiple arguments. We believe that concerning one argument at a time is more flexible and manageable, because such pairs can be applied independently, while their combination can cover many different relations.

We further categorize pattern templates into two groups: one with an explicit argument, and one with "null" argument. We will discuss pattern templates for argument realization in the next section, and then introduce methods to generate lexico-syntactic patterns. Lastly, we will give discuss pattern templates with null argument.

4.2.3.1 Pattern Templates for Argument Realization

Argument realization, which is at the heart of the area of linguistics, is the study of the possible syntactic expressions of the arguments of a verb [74]. In this study, we extend argument realization to nominal and adjectival triggers derived from verbs as well.

Pattern templates for verbal triggers

Below are examples of pattern templates for verbal predicate V_{tr} ("tr" stands for trigger) in active voice and passive voice.

Template 1 $NP_0\ VG$
$\langle VG\ head \rangle = V_{tr}$
$\langle VG\ head\ voice \rangle = active$
$\langle example \rangle =$ *"Runx3 binds"*

Template 2 $VG\ NP_1$
$\langle VG\ head \rangle = V_{tr}$
$\langle VG\ head\ voice \rangle = active$
$\langle example \rangle =$ *"expresses KBF1"*

Template 3 $NP_1\ VG$
$\langle VG\ head \rangle = V_{tr}$
$\langle VG\ head\ voice \rangle = passive$
$\langle example \rangle =$ *"OTF-1 is expressed"*

We use NP_1 in contrast to NP_0 in Template 1. This is because their roles are different. For example, in specification of Trigger 1, NP_0 is always the agent and NP_1 is always the theme.

In linguistics, verbal predicates can be further subcategorized by the presence and types of their syntactic arguments. Verbs that take just one argument are intransitive, while verbs with two and three arguments are transitive and ditransitive, respectively. In this study, we use the "frame" to capture the concept of trigger subcategorization and will discuss it in Section 4.2.4.1.

Pattern templates for nominal triggers

In addition to the standard pattern templates that are based on verbal forms of the trigger, we also consider cases where the trigger verb is nominalized (N_{tr}). For example, "transcribe" can be nominalized into "transcription" or "transcript". Nominalization of verbs can be divided into two classes. The first class is where resulting nouns denote actions, states, and processes. Their suffixes are typical "-ion", "-age", and "-ance" (e.g., "transcribe"→"transcription", "cleave"→"cleavage", and "appear"→"appearance"). The second class is where resulting nouns refer to entities (e.g., "transcribe"→"transcript" and

"produce"→"product"). We currently focus on the first class. Typical pattern templates for nominal triggers are:

Template 4 NP_{tr} *of* NP_1
$\langle NP_{tr}$ *head* $\rangle = N_{tr}$
$\langle$*example*$\rangle$ = *"expression of IFN-gamma"*

Template 5 $[\, NP_1\ NP_{tr}\,]_{NP}$
$\langle NP_{tr}$ *head* $\rangle = N_{tr}$
$\langle$*example*$\rangle$ = *"c-fos expression"*

Besides the theme, the agent can be incorporated via a "by" phrase. A pattern template for such instances is:

Template 6 NP_{tr} *by* NP_0
$\langle NP_{tr}$ *head* $\rangle = N_{tr}$
$\langle$*example*$\rangle$ = *"phosphorylation by Cdk5"*

As for non-directional relations/triggers we discussed earlier, we have additional templates, which are exemplified by the following:

Template 7 $[\, NP_{tr}$ *of* NP_0 - $NP_1\,]_{NP}$
$[\, NP_{tr}$ *of* NP_1 - $NP_0\,]_{NP}$
$\langle$*direction*$\rangle$ = *non-directional*
$\langle NP_{tr}$ *head* $\rangle$ = *non-directional trigger*
$\langle$*example*$\rangle$ = *"binding of p50 - p65"*

Pattern templates for adjectival triggers

English has a general morphological process of adjective conversion, which enables verbs to be used as adjectives. The pattern template for adjective triggers is Template 8 below. Adjectival derivations can be the present participle, the past participle, and the adjectivization of a verb. Currently we only implemented the past participle case as shown in Exampe 16.

Template 8 $[_{NP}\, ADJ\ \ NP_1\,]$
$\langle ADJ\ \ head\, \rangle = Adj_{tr}$
$\langle$*example*$\rangle$ = *"expressed pseudogenes"*

Example 16 a. the **transcribed** alpha globin gene

b. **phosphorylated** GSK3

4.2.3.2 Pattern Templates with Null Argument

There are cases when the writing style does not follow the common trigger-argument association. When the argument is omitted, but implied, we call them **elliptical construction**. Following are some examples of (a) elliptical constructions, and (b) how they would be written if the argument were not elided.

Example 17 a. When *phosphorylated*, **PI-1** inhibits PP-1.

b. When $\left[\text{PI-1}\right]_{\text{null argument}}$ is *phosphorylated*, **PI-1** inhibits PP-1.

Example 18 a. **GSK3** promotes p53 mRNA translation via *phosphorylation* of RNPC1.

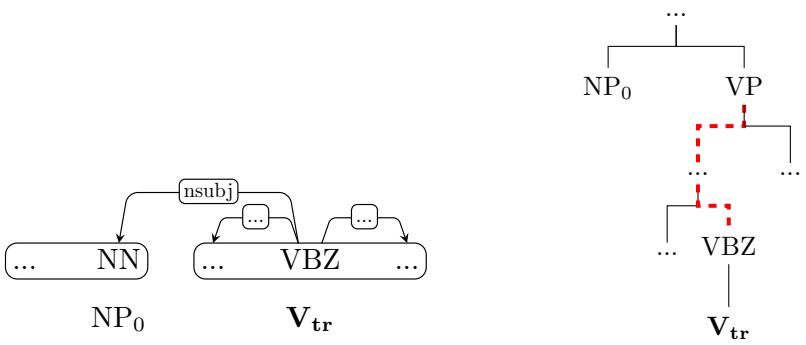
b. **GSK3** promotes p53 mRNA translation via $\left[\textbf{its}\right]_{\text{null argument}}$ *phosphorylation* of RNPC1.

$\Rightarrow$ RNPC1 is *phosphorylated* by $\left[\textbf{GSK3}\right]_{\text{null argument}}$.

Both (a) and (b) are grammatically correct and express the same underlying idea in Examples 17 and 18, but we tend to write (a) rather than (b). This situation is related to the null complement anaphora (deep anaphora) in a modern syntactic theory [126] and the implicit argument in a semantic theory [44]. For the relation extraction task, we observe that the elided argument may be found as its antecedent and determined by another trigger that selects it. Our framework recovers them as part of the relation extraction process, by applying for the null argument pattern templates. It should be noted that such elliptical constructions can appear in various positions of a sentence (e.g., at the beginning Exampe 17a or at the end Exampe 18a). These templates always rely on the whole sentence construct, therefore are too cumbersome to express. We designed some pattern templates such as Templates 9 and 10 to match sentences like Exampe 17a and Exampe 18a. Whether there exists a more general and clearer way to express these types of pattern templates needs to be further explored.

Template 9 *IN* VG_1, NP_1 VG_2 *NP*
$\langle VG\ head \rangle = V_{tr}$
$\langle VG\ head\ voice \rangle = passive$
$\langle example \rangle$ = *"When expressed in Arabidopsis, Tomato transcription factors pti4, pti5, and pti6 activate defense responses."*

Template 10 NP_0 *VG NP via* NP_{tr}
$\langle NP_{tr}\ head \rangle = N_{tr}$
$\langle VG\ head\ voice \rangle = active$
$\langle example \rangle$ = *"HSP90 inhibitors downregulates EGFR via phosphorylation at S1046/7"*



(a) Dependency graph (b) Parse tree

Figure 4.2: Sample representations of Template 1.

4.2.3.3 Representation of Pattern Templates

The specification of the template can depend on the syntactic representation. For example, Figure 4.2 shows two representations of Template 1 on Page 41. In this chapter, we use the constituency-based parse trees with tree regular expression [75] as the representation of pattern templates. To pick the head of phrases, we used Michael Collins' head table [30] (see red, dotted line in Figure 4.2b).

4.2.4 Lexico-syntactic Pattern Generation and Matching

Provided with a trigger specification, we use a collection of pattern templates to derive lexico-syntactic patterns specific to the trigger provided. We use "frame" to associate a trigger with a set of pattern templates. In the following subsections, we will first define frames then discuss how it can be combined with trigger specification and pattern templates to generate lexico-syntactic patterns.

4.2.4.1 Frames

A frame is a set of pattern templates sharing the same syntactic nature of the constituents that are likely to be associated with the trigger. It specifies the arguments of the trigger. We found that the most frequent frames in our use cases are:

Frame 1 NP_0/NP_1

$$\langle \textit{templates with } NP_0 \rangle = \{1, 10\}$$

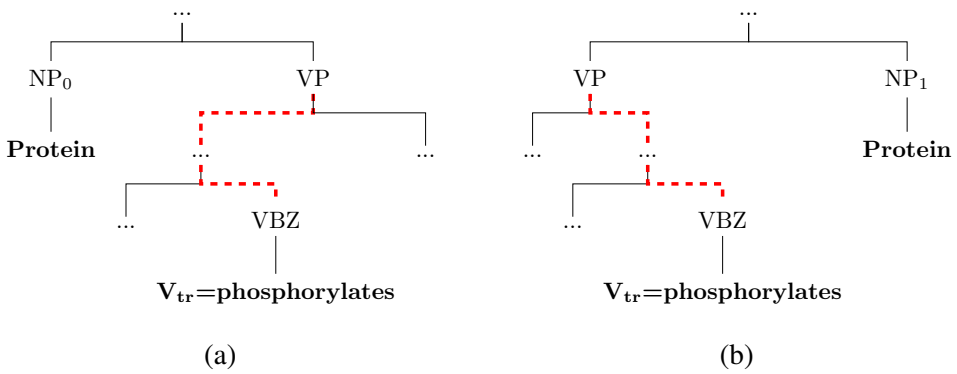


Figure 4.3: Lexico-syntactic patterns of Templates 1 and 2.

$$\langle \textit{templates with } NP_1 \rangle = \{2, 3, 4, 5, 8, 9\}$$
$$\langle \textit{templates with } NP_0/NP_1 \rangle = \{7\}$$

Frame 2 NP_0

$$\langle \textit{templates with } NP_0 \rangle = \{1, 10\}$$
$$\langle \textit{templates with } NP_0/NP_1 \rangle = \{7\}$$

Frame 3 NP_1

$$\langle \textit{templates with } NP_1 \rangle = \{2, 3, 4, 5, 8, 9\}$$
$$\langle \textit{templates with } NP_0/NP_1 \rangle = \{7\}$$

We distinguish NP_0 and NP_1, because semantically they play different roles and have different types in the trigger specification. Syntactically, they represent different grammatical constituents. To associate pattern templates with frames, verb type information is also used. For example, one of constraints in English is that only transitive verbs can be passivized.

4.2.4.2 Lexico-syntactic Patterns Generation and Matching

We present a case study utilizing trigger specification, pattern template, and frame to generate lexico-syntactic patterns.

Reconsider the specification of Trigger 1 for "phosphorylate" on Page 37. It specifies having Frame 1 which links to a total of eight pattern templates. Take Templates 1 and 2 as examples, we will automatically generate two lexico-syntactic patterns, as shown in Figure 4.3.

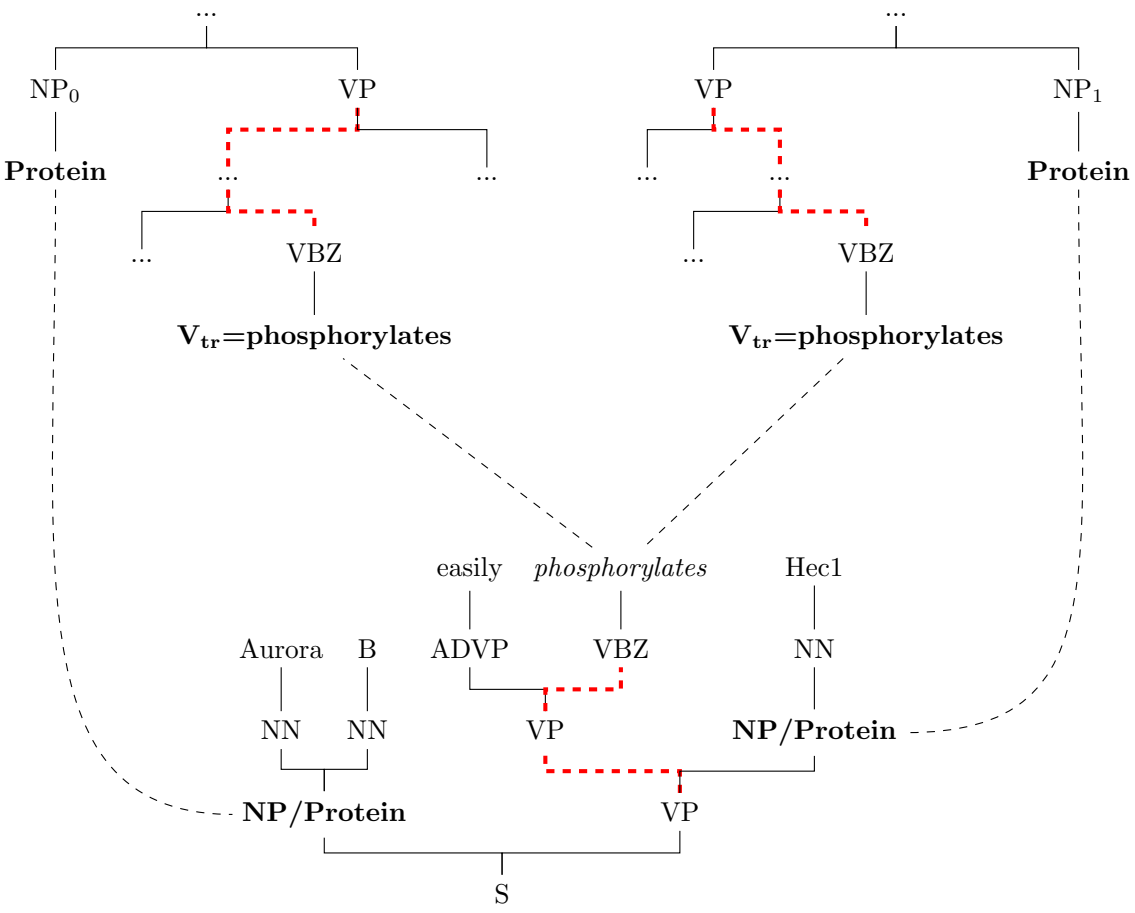


Figure 4.4: Parse tree of "Aurora B easily phosphorylates Hec1" (below) and it matches lexico-syntactic patterns in Figure 4.3.

The automatic generation procedure is similar to the concept of Lexical Tree Adjoining Grammars (LTAG). In LTAG, a tree family associates a verb lexeme with a given subcategorization. The subcategorization includes a set of grammatical structures that represent all the possible lexico-syntactic variations for that verb lexeme. So grammatical structures can be obtained by combining lexical rules and syntactic transformations. Compared with LTAG, the "frame" in our approach is essential but not exactly a subcategorization in LTAG. The trigger specifications are similar to tree families in LTAG, which associate a trigger lexeme with a given frame.

Consider the text fragment "$\left[\text{Aurora B}\right]_{\text{protein}}$ easily phosphorylates $\left[\text{Hec1}\right]_{\text{protein}}$", where "Aurora B" and "Hec1" have already been tagged as gene or gene product. Figure 4.4

illustrate its parse tree and how the tree matches lexico-syntactic patterns in Figure 4.3. Specifically, the two lexico-syntactic patterns in Figure 4.3 matches the "phosphorylates" word as a trigger, "Aurora B" as NP_0, and "Hec1" as NP_1. The trigger Trigger 1 checks the types of NP_0 and NP_1, which are proteins, and assign their roles for an agent and a theme. As a result, we get <phosphorylates$_{\text{trigger}}$, Aurora B$_{\text{agent}}$> and <phosphorylates$_{\text{trigger}}$, Hec1$_{\text{theme}}$>.

4.2.5 Sentence Simplification

So far, we have discussed how arguments can be extracted by matching patterns. But even with a large number of patterns automatically generated in the proposed manner, the recall of the resulting system is still low because sentence constructions and writing styles vary considerably in the actual text, and the number of variations to be considered is overwhelmingly high. Therefore, we introduce sentence simplification as a preprocessing module. Given an input sentence, the output of this module is a set of generated simplified sentences. For details of sentence simplification, please see Chapter 3.

4.2.6 Referential Relation Linking

By using patterns and sentence simplification, the system can detect textual expressions in the argument position. Sometimes, the referred entity is mentioned somewhere else in the text. Consider Exampe 19. The system can extract binding relation <dimerized$_{\text{trigger}}$, the protein$_{\text{theme}}$> from Exampe 19, but the actual target entity is "c-Fox" not "the protein". To link these phrases, we developed patterns to extract referential relations.

Example 19 The stability of **c-Fox** was decreased when **the protein** *was dimerized* with phosphorylated c-Jun.

4.2.6.1 Referential Relations

Referential relation patterns are designed to extract the relationship of one nominal phrase to another, when one provides the necessary information to interpret the other [48]. By resorting referential relations, an extraction system is able to identify an actual target entity beyond the initially extracted arguments.

Co-referential relations (or co-references) occur when multiple expressions refer to the same referent. For instance, in the previous example, "the protein" and "c-Fox" both refer to the same object. In a co-referential relation, the anaphoric reference can be a pronoun or definite noun phrase, and its antecedent can be the actual name of protein/gene.

Member-collection relations are useful in linking a generic reference to a group of entities that are specified in other places in text. Exampe 20 illustrates that the generic reference "adhesion molecules" can be extracted as an argument of the trigger "expression". Meanwhile, specific referred entities include "integrin alpha", "L-selectin", "ICAM-3" and "H-CAM". We consider patterns like "NP, *such as* NP (, NP)$*$" to identify this type of relations.

Example 20 expression of **adhesion molecules** *including* **integrin alpha, L-selectin, ICAM-3, and H-CAM**

Hyponymy relations are used when argument X is a hyponym of argument Y, if X is a subtype of Y, or when an instance of X refers to a concept Y. Thus, in Exampe 21a, "CD14" is said to be a hyponym of "membrane glycoprotein", and in Exampe 21b, "p130 Crk-associated substrate (Cas)" is a hyponym of "protein". When linked, the system extracts $<\text{expressed}_{\text{trigger}}, \text{CD14}_{\text{theme}}>$ and $<\text{phosphorylated}_{\text{trigger}}, \text{Cas}_{\text{theme}}>$, respectively. To achieve this goal, we identify the fragments having keywords such as "acts as" or "is identified as", which are similar to the ones in hearst1992automatic, snow2004learning. Moreover, the apposition construct can also hold a hyponymy relation between the appositive and the referred noun phrase.

Example 21 a. **CD14** *is a* membrane glycoprotein expressed specifically on . . .

b. **p130 Crk-associated substrate (Cas)** *was originally identified as* a highly phosphorylated protein.

Part-whole relations are useful when an argument extracted for a trigger comprises a part of the target entity. For biomedical information extraction, this framework focuses on relations between protein parts and a protein, e.g., a residue in a protein. *Part-whole*

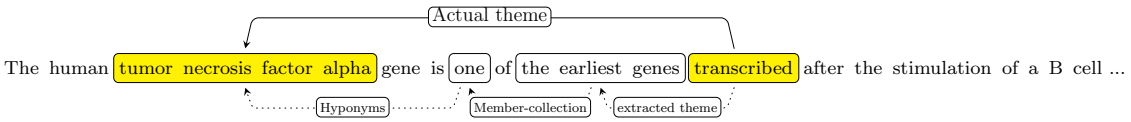


Figure 4.5: A sample referential relations linking.

relations in Exampe 22 can be captured by patterns such as "NP_{whole} *contains* NP_{part}" or by the existence of keywords such as "surface" and "domain".

Example 22 calcineurin *associated* with the N-terminal **domain** of $\left[\text{NFAT}\right]_{\text{protein}}$.

4.2.6.2 Linking Entities through Referential Relations

We will use the example in Figure 4.5 to illustrate integrating basic patterns and linking relations.

This example contains one transcription relation. Our goal is to extract its trigger and argument, namely <transcribed$_{\text{trigger}}$, tumor necrosis factor alpha$_{\text{theme}}$> which are highlighted in the sentence. We assume "tumor necrosis factor alpha" is typed as a gene.

Given the trigger "transcribe" and using Template 3 as discussed earlier on Page 41, we can extract <transcribed$_{\text{trigger}}$, the earliest genes$_{\text{theme}}$>. But "the earliest genes" is not an informative reference to the target entity. In addition, we extract one member-collection relation <one$_{\text{member}}$, the earliest genes$_{\text{collection}}$> and one hyponymy relation <tumor necrosis factor alpha$_{\text{hyponym}}$, one$_{\text{hypernym}}$>. The first relation enables us to infer <transcribed$_{\text{trigger}}$, one$_{\text{theme}}$>, since the collection of genes ("the earliest genes") are "transcribed" and, then, one of its members can be "transcribed" as well. Then, the latter relation allows us to state "tumor necrosis factor alpha" is the "one" in this context and hence to conclude <transcribed$_{\text{trigger}}$, tumor necrosis factor alpha$_{\text{theme}}$>.

The Algorithm 2 for the linking procedure is as follows: First we collect all referential relations in the document. Then, for every instance matched with extraction patterns for a trigger, if the instance's argument is not an informative reference, we recursively search for all of its references in the detected referential relations. If an appropriate reference of an entity is found, we link it to the trigger, by creating a new pair <trigger, referred entity>. This search

procedure ends when we exhaust all possibilities. As a result, more than one pair may be created and all pairs are proposed.

Algorithm 2 Algorithm for linking entities through referential relations.

```
1:  procedure LINKING(i, rls)  ▷ link matched instance i based on referential relations rls
2:      q ← an empty Queue
3:      result ← {}
4:      ENQUEUE (q, i)
5:      while q is not empty do
6:          i ← DEQUEUE(q)
7:          if ARGUMENT[i] is a protein then
8:              result ← result ∪ {i}
9:          else
10:             find r in rls where ARGUMENT[i] = REFERER[r]
11:             if r ≠ Nil then
12:                 ENQUEUE(q, <TRIGGER[i], REFEREE[r]>)
13:             end if
14:         end if
15:     end while
16:     return result
17: end procedure
```

4.3 A GE Relation Extraction System

Our framework is designed to extract a variety of relations. We wish to evaluate our framework by considering the extraction of different types of relations. Furthermore, the data set should include trigger annotations needed to automatically generate patterns. For these reasons, we chose to use the corpus of BioNLP-ST (Shared Tasks) 2011 GE task, which included several events/relations extraction subtasks.

4.3.1 System implementation

The raw text was parsed by Charniak-Johnson parser using David McClosky's biomedical model [129]. We did not have to use a full parser, but we chose Charniak-Johnson parser because it is arguably the best on the GENIA Treebank and PubMed abstracts [81, 130], and also because it was convenient in comparing the evaluation with existing systems. But other parsers would also work with little integration effort.

Typing is critical and presumed. We noticed at least three important ways of using typing in the system: (1) for relations like phosphorylation, the theme needed to be a noun phrase of type protein or protein part; (2) for the transcription relation, the theme needed to be of type mRNA; and (3) for the binding relation, one of its themes needed to be a protein or protein part.

To obtain the argument type, we used a modified version of BioNex [91], which was developed based on ideas from [90] and used in RLIMS-P [50]. It can detect semantic types of entities referred by nouns/noun phrases, such as protein/gene/chemical or their part. For detecting RNA type for transcription, we relied only on the NP's head word (mRNA or transcript).

Patterns were generated and matched from the parse tree using tree regular expression [75]. Thus, pattern templates were designed using tree regular expression as well. 26 pattern templates were created.

For the simplification task, we implemented iSimp [101] on full parsed trees, and generated all possible simplified sentences (with its parsed trees) for pattern matching. This enabled us to quickly implement the system easily and compare its results with and without using the sentence simplification module. For the coreference detection, we extended the ideas used in RLIMS-P. Other referential relation patterns were defined using tree regular expressions.

4.3.2 Evaluation

4.3.2.1 BioNLP-ST 2011 GE Task

The BioNLP-ST GE task series aim to provide the community with shared resources for the development and evaluation of fine-grained information extraction (IE) systems [62]. Each time, it was organized with a grand theme (a goal shared by all the tasks): introduction of the event extraction task, generalization, and knowledge base (KB) construction, for the 1st, 2nd and 3rd editions, respectively [58, 59, 93]. For the purpose of evaluating the generalizability of the framework, we used BioNLP-ST 2011 GE task corpus to evaluated the relation extraction system in this dissertation. We will refer to them as "GE task" and "GE

Table 4.1: Statistics of of the datasets for the BioNLP-ST 2011 ST GE task.

Item	Training		Devel		Test	
	Abstract	**Full**	**Abstract**	**Full**	**Abstract**	**Full**
Articles	800	5	150	5	260	4
Words	17,6146	29,583	33,827	30,305	57,256	21,791
Proteins	9,300	2,325	2,080	2,610	3,589	1,712
Simple Event	2,858	657	559	549	1,186	385
Gene_expression	1,738	527	356	393	722	280
Transcription	576	91	82	76	137	37
Protein_catabolism	110	0	21	2	14	1
Phosphorylation	169	23	47	64	139	50
Localization	265	16	53	14	174	17
Binding	887	101	249	126	349	153
Total	*3,745*	*758*	*808*	*675*	*1,535*	*538*

corpus" hereafter. In GE task, evaluation results were reported on (W)hole, (A)bstract, and (F)ull paper collections, respectively [60]. The abstract collection contains paper abstracts, the full-text collection contains full papers, and the whole collection contains both abstracts and full text. Following the same setting, we also report our results on W, A, and F.

GE corpus covers nine types of events: *Gene_expression, Transcription, Localization, Protein_catabolism, Phosphorylation, Binding, Regulation, Positive_regulation, and Negative_regulation*. Among these, we focused on events with simple entities as themes. Thus, Regulation and its subtypes were removed because their themes could be other events with other triggers. As a result, only the first 6 types of events were evaluated. The first five events were called "Simple Event" collectively [60]. Table 4.1 shows statistics of training, development and test sets for the GE tasks used in this evaluation.

4.3.2.2 Trigger Selection

Since our approach requires a list of triggers, we used the triggers annotated in the corpus. To effectively evaluate our framework, we further decided to focus on a selected group of triggers. Among triggers in GE corpus, we chose only the triggers that are based on verbs (e.g., phosphorylate) and their nominal and adjectival forms (Table 4.2) as discussed

Table 4.2: Selected triggers from the training set of BioNLP-ST 2011 GE task. The *Derivation* column shows affix used to derive other forms of triggers. Singular, past tense, and gerund forms are not shown.

Events	**Verb**	**Derivation**
Gene_expression	express	-ion, over-, co-, non-, re-
	produce	-ion, non-, co-
Transcription	express[1]	*see above*
	initiate	-tion
	produce[1]	*see above*
	transcribe	-tion, -tional, -tionally
Protein_catabolism	cleave	-age
	degrade	-tion, -tive
	proteolyse	-sis, -tic, -tically
Phosphorylation	phosphorylate	-ion, under-, hyper-
Localization	accumulate	-ation
	appear	-ance
	detect	
	export	
	express[2]	*see above*
	import	
	localize	-ation, co-
	migrate	co-
	mobilize	-ation, im-
	release	
	secrete	-ion
	translocate	-ion
Binding	associate	-ion
	bind	DNA-
	interact	-ion
	ligate	-ion, co-
	link	cross-
	oligomerize	-ation

1. This predication is always used together with “mRNA”.
2. This predication is always used together with “surface”.

Table 4.3: Statistics of events with selected triggers on BioNLP-ST 2011 ST GE task. For binding events, we enable event decomposition mode. If an event's argument is within an equivalence relation with n members, this event will be counted n times. % = Events with selected triggers/All events.

Events	**Training set**		**Devel set**	
Simple Event	3,165	84.92 %	923	80.19 %
Gene_expression	2,094	86.64	614	79.23
Transcription	511	72.59	115	69.28
Protein_catabolism	105	92.11	22	95.65
Phosphorylation	185	94.87	107	95.54
Localization	270	90.91	65	86.67
Binding	874	71.00	380	75.55
Total	*4,039*	*81.46 %*	*1,303*	*78.78 %*

before. We did not use the triggers that are pure nouns (e.g., level) or adjectives (e.g., positive). Additionally, we eliminated verb triggers like "find" and "form" because they are not specific to particular biomedical events.

After trigger selection, events related to the selected triggers were found to be very frequent in the corpus, covering 81.46% and 78.78% of all events in the training and development sets of the GE corpus, respectively (Table 4.3). In other words, the upper bound of recalls are 81.46% and 78.78%. Given the assumption that the precision is 100%, our system's upper bound of F-scores are limited to 89.78% and 88.13% respectively.

4.3.2.3 Evaluation Measurement

Following the same evaluation criteria of the GE task evaluation, we considered event decomposition. This means that Binding events with multiple primary arguments are decomposed into multiple single primary argument events, and are treated as separate events in the evaluation. We tested our framework on both the training and development sets of the GE corpus, where we used only the protein annotations. The evaluation was carried out by comparing the predicted annotation with the gold standard. We used the same strict matching mode as in the GE task [60], which requires extracting equality between the two events/relations as follows:

1. the event/relation types are the same;
2. the triggers are the same; and
3. the arguments are the same.

Same triggers and arguments mean that their text spans are the same (i.e., two text spans, (a_1, b_1) and (a_2, b_2), are the same iff $a_1 = a_2$ and $b_1 = b_2$).

4.3.2.4 Results

Table 4.4 summarizes the performance of our system on the development set of the GE corpus. We provide results for the Simple Event averaging over five events, results for each of the six individual events including Binding, as well as the overall results for all events. Overall, we obtained a global F-score of 71.47%.

The second part of the results shows the Precision/Recall/F-score when we limited the task to subset events containing only selected triggers. Here, we achieved an F-score of 81.36%, with a higher precision and a higher recall.

Table 4.5 shows the effects of different system components on the overall results of our system. We considered three scenarios: (1) using only the argument and null argument patterns, (2) using also the sentence simplification, and (3) using both sentence simplification and referential relation linking. Overall, sentence simplification increased the recall by 22%, while referential relation linking achieved an addition 7% increase. Results indicated that without increasing the number of patterns, simplification, and referential relation linking are helpful in extracting more instances of relations.

Table 4.6 summarizes the performance of our system on the test set of the GE corpus using the online evaluation system[1] by the approximate span & recursive evaluation method. Overall, we obtained a global F-score of 72.16% for Simple Event, and 50.50% for Binding. The best overall performance on Task 1 in BioNLP-ST 2011 achieved an F-score of 73.90% with Simple Event, and 48.49% with Binding event. The best rule-based system (ConcordU

[1] http://bionlp-st.dbcls.jp/GE/2011/eval-test/

Table 4.4: Evaluation results on the Whole, Abstract, and Full paper collections from the development set of BioNLP-ST 2011 GE task. Performance is reported in terms of Precision, Recall and F-score.

	Whole			**Abstract**			**Full**		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Whole set									
Simple Event	92.06	63.42	75.10	92.04	65.61	76.61	92.08	61.05	73.42
Gene_expression	92.28	64.77	76.12	91.01	66.75	77.02	93.61	62.88	75.23
Transcription	89.13	49.40	63.57	94.55	57.78	71.72	81.08	39.47	53.10
Protein_catabolism	94.12	69.57	80.00	93.75	71.43	81.08	100.00	50.00	66.67
Phosphorylation	98.77	71.43	82.90	96.77	62.50	75.95	100.00	78.13	87.72
Localization	84.75	66.67	74.63	91.49	70.49	79.63	58.33	50.00	53.85
Binding	91.51	47.12	62.20	86.96	42.94	57.49	98.98	54.80	70.55
Total	*91.92*	*58.46*	*71.47*	*90.65*	*57.62*	*70.46*	*93.53*	*59.53*	*72.76*
Subset with selected triggers									
Simple Event	91.17	78.33	84.27	91.10	78.74	84.47	91.26	77.86	84.03
Gene_expression	91.18	80.78	85.66	89.93	82.51	86.06	92.48	79.10	85.27
Transcription	89.13	71.30	79.23	94.55	73.24	82.54	81.08	68.18	74.07
Protein_catabolism	94.12	72.73	82.05	93.75	71.43	81.08	100.00	100.00	100.00
Phosphorylation	98.77	74.77	85.11	96.77	65.22	77.92	100.00	81.97	90.09
Localization	83.05	75.38	79.03	89.36	79.25	84.00	58.33	58.33	58.33
Binding	90.73	61.84	73.55	85.71	57.02	68.49	98.98	70.29	82.20
Total	*91.06*	*73.52*	*81.36*	*89.63*	*71.60*	*79.61*	*92.89*	*76.01*	*83.61*

Table 4.5: Comparative results of subset events with selected triggers on the Whole, Abstract, and Full paper collections from the development set of BioNLP-ST 2011 GE task. Performance is reported in terms of Precision, Recall and F-score. "Basic patterns" = using pattern templates for argument realization and pattern templates with null argument to generate patterns.

	Whole			**Abstract**			**Full**		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Basic patterns									
Simple Event	93.01	51.90	66.62	91.67	48.99	63.85	94.42	55.24	69.71
Binding	94.95	24.74	39.25	91.67	22.73	36.42	100.00	28.26	44.07
Total	*93.32*	*43.98*	*59.78*	*91.67*	*40.35*	*56.04*	*95.17*	*48.68*	*64.41*
Using sentence simplification									
Simple Event	92.99	74.76	82.88	92.71	74.70	82.74	93.31	74.83	83.05
Binding	94.59	46.05	61.95	91.67	45.45	60.77	100.00	47.10	64.04
Total	*93.31*	*66.39*	*77.58*	*92.47*	*65.08*	*76.40*	*94.38*	*68.08*	*79.10*
Using sentence simplification and referential relations									
Simple Event	91.17	78.33	84.27	91.10	78.74	84.47	91.26	77.86	84.03
Binding	90.73	61.84	73.55	85.71	57.02	68.49	98.98	70.29	82.20
Total	*91.06*	*73.52*	*81.36*	*89.63*	*71.60*	*79.61*	*92.89*	*76.01*	*83.61*

in Table 4.6) achieved F-scores of 70.52% and 36.88% with Simple and Binding events, respectively.

For Simple Event extraction, we observed that our F-score is lower than FAST and UMass. This is because though we achieved the highest precision, the recall is much lower. We compared with the results on the development set, we can see the same phenomenon: while the precision stays same, there is a drop on recall. These observations indicate there are some cases that failed to be matched by rigid patterns. On the other hand, we also observed that ConcordU achieved the similar recall on the test set. This might also imply that these missing errors are common among rule-based systems. Since we cannot obtain the annotations of the test set, no further analysis is possible. For Binding Event extraction, the recall was also dropped. But we gained a significant improvement on the precision, therefore the F-score is higher than other systems.

Table 4.6: Comparative results on the Whole paper collections from the testing set of BioNLP-ST 2011 GE task 1. Performance is reported in terms of Precision, Recall and F-score.

Team	Simple Event			Binding		
	P	**R**	**F**	**P**	**R**	**F**
FAST	80.25	68.47	73.90	44.20	53.71	48.49
UMass	67.01	81.40	73.50	56.42	42.97	48.79
ConcordU	85.53	59.99	70.52	49.66	29.33	36.88
Ours	*92.34*	*59.22*	*72.16*	*90.18*	*35.07*	*50.50*

4.4 miRTex: a miRNA-gene Relation Extraction System

MicroRNAs (miRNAs) are an important class of RNAs that regulate a wide range of biological processes by post-transcriptional regulation of gene expression. The framework discussed in this chapter was used to develop a text mining system, miRTex, that extracts miRNA-gene regulation, miRNA-target, and gene-miRNA regulation relations [76].

The Stanford sentence splitter [79], and a well-known name detector, BANNER [70] (for gene mention) were used. We used an in-house rule-based detector to recognize the miRNA mentions. Following name detection, parse trees of the sentences are generated to match with our lexico-syntactic rules. We used the Charniak-Johnson parser with David McClosky's biomedical model [23, 129].

The set of trigger words is formed by the verbs and their nominal and adjective forms in the most frequently used trigger words indicating regulation relation from the BioNLP 2013 GE corpus, and the trigger words seen in our development corpus [61]. The verbs are either words that indicate regulation, such as "regulate", "increase", "mediate" and "suppress", or specific action verbs that indicate how a miRNA interacts with a gene product, such as "target", "cleave" and "bind".

As described in the framework, the rules to extract regulation relations contain a syntactic pattern with typing constraints. Because we are interested in the regulation of a gene by a miRNA, typing constraints require that the agent must be a miRNA and the theme must be a gene. To extract gene-miRNA regulation relations, the only necessary modification is to switch the typing constraints of the agent and theme.

miRTex was evaluated on two different corpora. On an in-house test set containing 150 abstracts, miRTex obtained the precisions, recalls, and F-scores of miRNA-gene, gene-miRNA, and miRNA-target extraction 96/91/94, 94/83/88, 96/81/88 respectively. On the test set of Bagewadi et al.'s corpus [6] that contains 100 abstracts, miRTex gave the performance of 87% F-score with 92% precision and 82% recall for the 123 miRNA-gene associated pairs.

We showed that by implementing the framework discussed in this chapter, the elaborate use of lexico-syntactic information and linguistic generalizations enables miRTex to achieve the state-of-the-art performance.

4.5 Conclusions

In this chapter, we have designed a framework for rapid development of biomedical relation extraction systems. The framework requires as input only a list of triggers and their specifications to retrieve relations of interest. These are used to automatically generate lexico-syntactic patterns, by making use of linguistic theories. In applying extraction patterns, the framework uses sentence simplification and referential relations to improve the performance, especially the recall while maintaining the precision.

To evaluate the performance of the framework, we implemented two text mining systems.

We produce a relation extraction system and evaluate it on the BioNLP-ST 2011 GE task. The system achieved F-scores of 71.47% on GE development set. For a specific subset of examples whose triggers are limited to verbs or verb derivatives, we obtained F-scores of 81.36%. On the testing set, our results are favorably comparable with state-of-the-art systems as well. The results are consistent with our hypothesis that we can achieve good precision and recall with the range of patterns generated from triggers and that simplification and referential relation linking serve to increase the recall while maintaining the precision. The system was produced using general resources and the only aspect specific to BioNLP-ST 2011 GE task was the selection of trigger words that appear in the training corpus. Except for the specification of triggers, other aspects (parser, typing system, simplification, pattern matching system) are general purpose systems that we have already used in our existing

systems. This meets the desired usage of the framework where the only expected input required from domain experts is the specification of the triggers. This, together with the fact that no training set is required, is one of our motivations for developing the framework: ability to create effective relation extraction systems for new relations where resources (information of annotated corpus) are not publicly available.

Another text mining system that was developed using our approach is miRTex for extraction of miRNA-target relations as well as miRNA-gene and gene-miRNA regulation relations. The resulting rule-based system achieves high recall while preserving the high precision. The system achieved the state-of-the-art performance on a test set of 150 abstracts with the evaluation results showing that the precision of our system greatly outperforms the co-occurrence-based method with a comparable recall. The text mining results for all the Medline abstracts are stored in a database that can be searched through the website at http://proteininformationresource.org/mirtex.

Chapter 5

EXTENDED DEPENDENCY GRAPH

This chapter continues in the directions of previous chapters but extracts relations by (1) using parsers that have improved considerably in capturing syntactic structures, (2) going beyond just syntax in the representation of sentences, and (3) producing output representation that can be equally accessible to rule-based and machine-learning based relation extraction systems.

In this chapter, we propose Extended Dependency Graph (EDG) to capture relations between words and phrases in a sentence. EDG not only considers syntactic dependencies between words in a sentence, but also utilizes information beyond syntax to capture different dependencies. We believe the use of EDG will enable machine-learning and rule-based methods to generalize more easily. Experiments confirm that (1) EDG provides up to 10% f-score improvement over dependency graph using mainstream kernel methods over five corpora. We conducted additional experiments to provide a more detailed analysis of the contributions of individual modules in EDG construction, and (2) EDG with a few rules enables us to get good coverage of GE event detection on BioNLP 2011 GE task corpus. We are able to obtain f-scores of 74.86% and 70.31% on the development and test sets, which lead to 3.4 and 1.1 percentage points increase over the system in Chapter 4 respectively.

5.1 Introduction

Similar to what is discussed in Chapter 4, we are trying to alleviate the issues of dealing with textural variations in relation extraction systems. In the biomedical domain, most relation extraction work is currently applied on the abstracts of articles. These abstracts by nature are dense with information and often use constructions such as appositives and relative

clauses. The abundance of textual variations can thus be problematic for biomedical relation extraction systems, especially with small training corpora.

One solution to this issue is to find a suitable level of abstraction in the text representation so that either machine-learning methods generalize more easily, or simple extraction patterns can match a good coverage of relations in the text. Use of syntax and parse information provides one such abstraction and has become prevalent in biomedical relation extraction. It has been suggested dependency links are close to the semantic relationship needed for the next stage of interpretation [33].

There have been significant advances in the development of advanced machine learning methods and the use of sophisticated rules in the biomedical domain. In this chapter, we focus on the representation of the text used in learning (or rules) rather than the machine-learning technique (or pattern generation), with the hope that advances in both directions will improve the performance of the relation extraction systems. We propose Extended Dependency Graph (EDG), which includes information about text that goes beyond syntax. We will define EDG and discuss how we construct it from a given sentence by using some simple linguistic notions.

The hypothesis we test here is that EDG allows ML techniques (or patterns) to generalize more easily. To determine the effect of EDG, we conducted two experiments. We tested a machine-learning system on protein-protein interaction (PPI) extraction. For this purpose, we used two kernels: a simple kernel based on edit distance [40] and a more elaborate kernel that is one of the top performing kernels on the PPI task [3]. We compared the performance of both kernels using dependency graph and EDG on 5 corpora. Our results suggest EDG provides up to 10% f-score improvement over dependency graph. On 3 out of 5 corpora, the results are better than the overall best system in the study of [132], as well as an ensemble method that builds on them [84]. We also evaluate the contributions of the individual components included in EDG.

In order to evaluate the utility of EDG in constructing rule-based systems, we tested a rule-based system that uses EDG on the BioNLP 2011 GE task which contains six relations. The system with EDG extends the framework for fast development of pattern-based biomedical relation extraction (see Chapter 4), as well as the work that applies sentence simplification

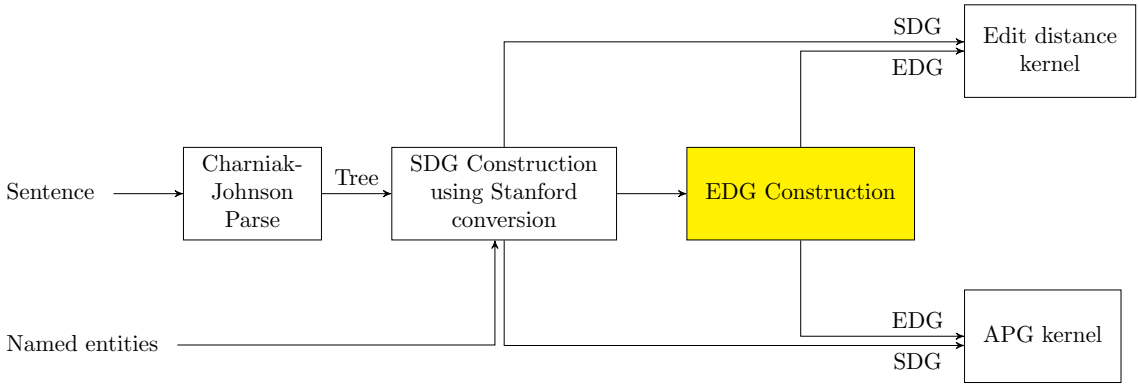


Figure 5.1: The framework of Extended Dependency Graph construction.

to improve the coverage of extracted relations (see Chapter 3). By using EDG, we can achieve better results than our previous system (iXtractR). Moreover, EDG enables us to greatly reduce the number of matching patterns. This indicates a deep analysis of text can be combined with the gains obtained from using various techniques and resources.

5.2 Method

Figure 5.1 illustrates the overall architecture with the core component highlighted: EDG construction. The input is a sentence with named entities marked. We use Charniak-Johnson parser and Stanford conversion tool to get the basic syntactic dependency graph (SDG). Our approach focuses on how to leverage simple linguistic principles and information beyond syntax to construct EDG from SDG.

5.2.1 Extended Dependency Graph (EDG)

We use EDG to represent the structure of the sentence. Like in the case of many dependency graph representations used in relation extraction, the vertices in an EDG are labelled with information such as the text, part-of-speech, and the word lemma. If an entity mention spans multiple tokens in a sentence, we merge their corresponding vertices (called contracting vertices) into one vertex.

EDG has two types of dependencies. The syntactic dependencies that are obtained from collapsed dependencies output by applying Stanford dependencies converter on a

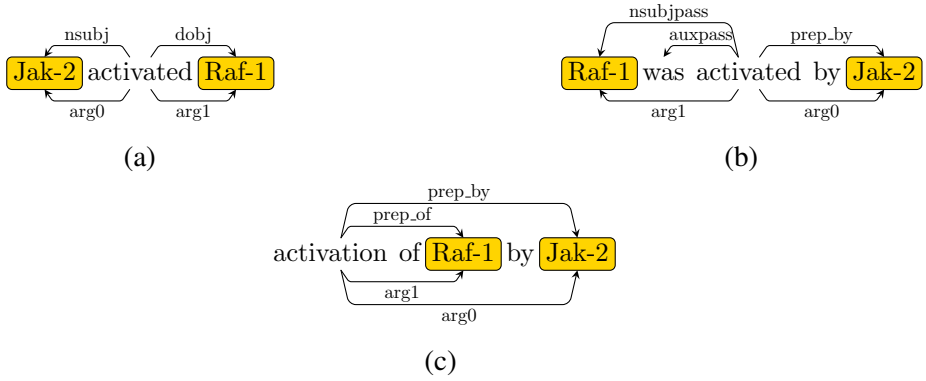


Figure 5.2: Sample EDGs with an active (a), passive (b), or normalized (c) verb.

syntactic parsing tree [35]. We introduce another type of dependencies, called the numbered arguments, which is based on the guidelines of PropBank [11]. Because we are currently focusing on binary relation extraction, we use only *arg0* and *arg1* (probably better stated as not-*arg0*) in EDG. Figure 5.2 shows EDGs of three text fragments with syntactic edges appearing above the words and numbered argument edges appearing below. While the syntactic dependencies vary in Figure 5.2, from a relation extraction perspective they are less relevant. In contrast, their numbered arguments between the lexeme "activate" (corresponding to words "activated" and "activation") and the two protein arguments are same.

There are two motivations for using numbered arguments. One is to "provide consistent argument labels across different syntactic realizations of the same verb" [11]. This is introduced for purposes of making generalizations easier downstream. The other is to add/propagate new *arg0* and *arg1* using the reasoning that goes beyond syntax.

We will now discuss how to capture *arg0* and *arg1* using different syntactic dependencies obtained from Stanford dependencies. Then we will describe relations such as *is-a*, *member-collection*, and *part-whole* and how to propagate *arg0* and *arg1* based on this relations.

5.2.2 Syntax Based *arg0* and *arg1*

We follow approaches of SemRep [115] and PASMED [95] to obtain the basic edges *arg0* and *arg1* from the syntactic dependencies. For example, EDG will include an *arg0* from

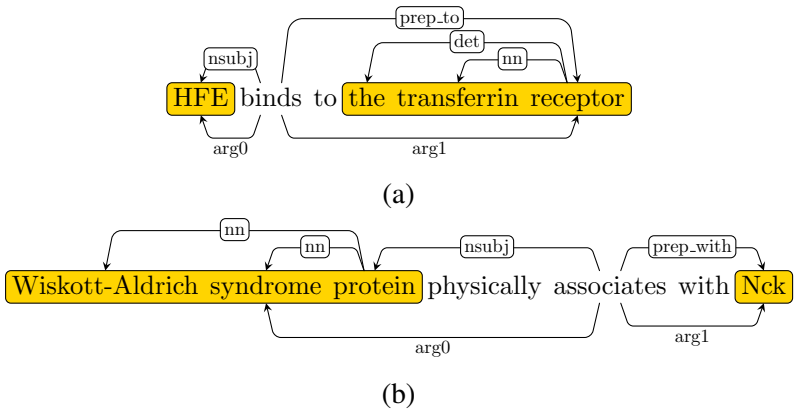


Figure 5.3: Sample EDGs with a verb and a prepositional phrase.

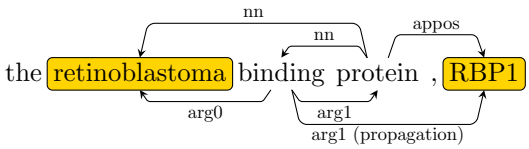


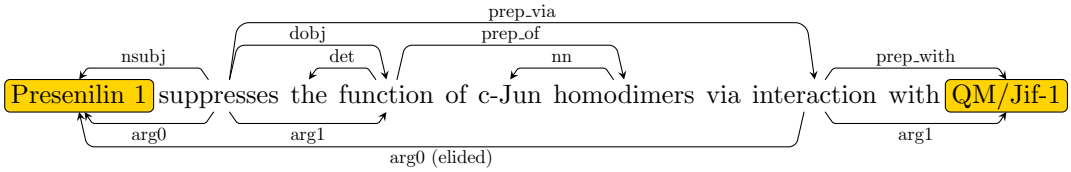
Figure 5.4: A sample compound noun phrase.

a verb to the noun if the syntactic dependency is *nsubj* or *agent* and include an *arg1* if the dependency is *nsubjpass* or *dobj*. To deal with intransitive verbs, we use edges like *prep_to* and *prep_with*, as shown in Figures 5.3a and 5.3b. An intransitive verb has no direct object, but it can be modified by a prepositional phrase to describe the action in detail. In this case, the prepositional phrase and the verb constitute *arg1*.

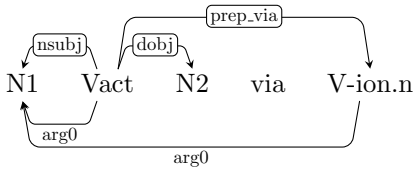
In addition, we consider the situation where verbs in gerund form are used as noun modifiers. Figure 5.4 shows a compound noun phrase. We know that there is a PPI between "retinoblastoma" and "protein", because we can rewrite the phrase into "retinoblastoma binds to protein, RBP1". Therefore, we add *arg1* from "binding" to "protein" in Figure 5.4. This operation will introduce cyclicity because the gerund is included in the noun phrase headed by "protein", which causes a SDG edge from "protein" to "binding". These edges are useful when found in combination with other constructions, such as appositive. We will discuss how to propagate *arg1* from the gerund "binding" to "RBP1" later.

Next we consider two cases of argument elision.

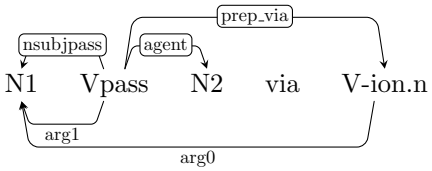
Elided argument relation Here we consider cases when the argument of a predicate



(a) A sample elided argument relation.



(b) Patterns with active form of the verb.



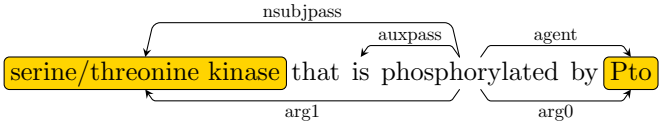
(c) Patterns with passive form of the verb.

Figure 5.5: Elided argument relation examples and patterns.

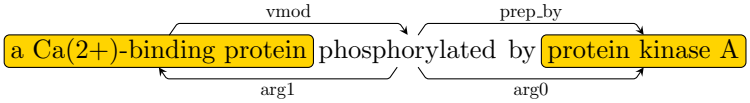
is not explicit but implicit. Figure 5.5a shows a sentence where *arg0(interaction, Presenilin 1)* can be inferred. The SDG includes a *prep_via* from the first verb "suppresses" to the nominalized verb "interaction", to indicate the PP attachment to the verb. In this case, we add an edge *arg0* from the nominalized verb to the *arg0*-argument of the first verb. Figures 5.5b and 5.5c demonstrate the pattern to generate the numbered argument edges, with active and passive voices of verbs respectively. In general, the nominalized verb ("V-ion") share the same subject with the main verb ("Vact" or "Vpass"). Thus we only add edges between "V-ion" and "N1". In constructing EDG, we also consider *prep_through* as well as *prep_by* when a gerund verb, rather than a nominalized form of a verb, follows it.

Reduced relative clauses Relative clause is a clause that modifies a noun phrase. There are two types of relative clauses that frequently appear in the biomedical text. Full relative clauses are introduced by relative pronouns, such as "which" and "that". Reduced relative clauses start with a gerund or past participle and have no overt subject.

The PropBank annotation guidelines [11] posit a numbered argument link from the relative clause verb to the trace in the parse tree which also indicates the referent noun phrase. For full relative clauses, we follow the normal procedure for verbs (Figure 5.6a). For reduced relative clauses, since we use the dependency structure that includes no traces, we use the edge *vmod* in the SDG from the head of the noun phrase to the reduced relative clause's verb (Figure 5.6b). The direction of this edge indicates that the relative clause is syntactically



(a) A sample full relative clause.



(b) A sample reduced relative clause.

Figure 5.6: Sample relative clauses.

included in the larger noun phrase. For the arg edge, we reverse the direction of *vmod* and create an edge from the relative clause's verb, as shown in Figure 5.6b. When compared to Figure 5.6a, the *arg* construction unifies the treatment for full relative clauses, in contract to the dependencies at the syntactic level.

5.2.3 Going Beyond Syntax

Here we consider the propagation of *arg* using information that goes beyond syntax.

Co-reference If an edge *arg* from a vertex *v* reaches a pronominal node, we add a new edge *arg* from *v* to any named entity the pronoun co-refers to. To detect the coreference we use the implementation of the technique described in [112]. For the acronyms with long-form and short-form, we treat them in the same way as coreference. We add extra edge *arg* when there is an *arg* incident on the long-form. We use the acronym detector of [120] to add acronyms missed in SDG. Interestingly, SDG uses *appos* for both acronym and appositive.

Appositive Reconsider the fragment "the retinoblastoma binding protein, RBP1" in Figure 5.4. Using the construction discussed thus far, the *arg1* will reach "protein". Further, SDG uses an edge *appos* from "protein" to "RBP1" for appositional modifier. We integrate *arg1* and *appos* to construct another edge *arg1* from "binding" to the actual named entity "RBP1". Sometimes, there is no comma in the apposition structure such as "two thyroid proteins Ht 21 and Ht 31". In these cases, *dep* is often used in SDG, from "protein" to "Ht 21" in the above example. When constructing EDG, we also consider these situations and correct SDG as necessary.

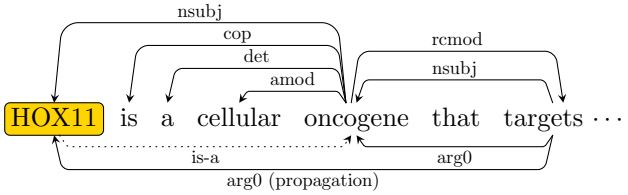


Figure 5.7: A sample *is-a* relation.

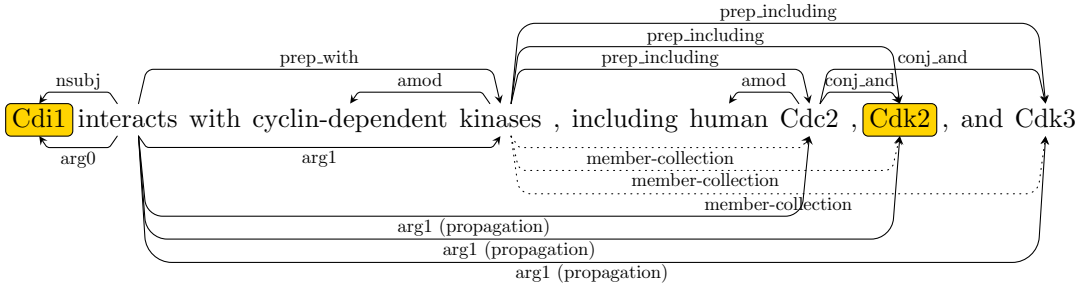


Figure 5.8: A sample *member-collection* relation.

Is-A In addition to appositive, we consider other forms of *is-a* relation mentioned textually, but cannot be directly found from the syntactic dependencies. For example, in Figure 5.7, there is no edge in SDG to explicitly capture the *is-a* relation. It is worth noting that the edge *nsubj* itself does not indicate the *is-a* relation, but together with two other edges *cop* and *det*, we can figure it out. Hence we add a new edge from "oncogene" to "HOX11" to reflect this relation in EDG (dotted edge). Afterward, we propagate *arg0* from "targets" to "HOX11".

Besides the pattern shown in Figure 5.7, we also identify "known as", "designated as", "considered as", "identified as" and "act as" as patterns that signal *is-a* relations. These patterns contain and extend rules in [49, 127].

Member-collection links a generic reference (called **collection**) to a group of entity mentions (called **members**). Like in Figure 5.8, typical key words that can identify *member-collection* relations are "including" and "such as". We consider the cases where mention group follows the keywords and the generic reference precedes these words. After the detection, we propagate *arg* from the collection to its members.

Part-whole links an entity part to its mention, typically denoting construction of larger entities out of smaller ones. Just like "breaking the glass of the window" can be stated as

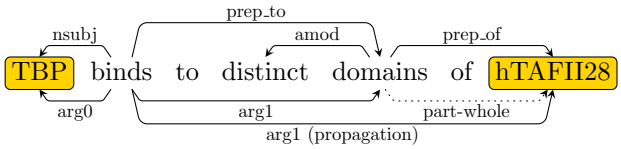


Figure 5.9: A sample *part-whole* relation.

"breaking the window", in biomedical tasks an action on a larger unit can often be inferred from a mention of the action applied on its part. That is, in Figure 5.9, after we detect a *part-whole* relation, an edge *arg1* incident on the part is propagated to the object that contains it.

In this paper, we focus on three types of patterns to recognize *part-whole* relations. The first is the preposition phrase such as "domain of *e*". Here "domain" indicates the part and *e* indicates the larger entity mention the "domain" belongs to. Other keywords indicating parts include "fragment", "portion", and "region". The second structural elements is a compound nominal like "*e* domain". The third group exploits keywords such as "contain", "consist", and "compose". For each *part-whole* relation, we propagate edges from the part to its entity mention.

5.3 Evaluation

We evaluated our method on protein-protein interaction (PPI) extraction task, where the system identifies whether a given protein pair in a sentence has PPI relationship or not. We used SDG or EDG as input representation of the sentences, which includes the named protein entities.

5.3.1 Evaluation of Kernel-based Systems

5.3.1.1 Kernels

We tested the effect of using EDG on two kernels that have been employed for PPI extraction.

Edit distance kernel is based on the edit distance among the shortest paths between entities in the dependency graph and is based on the minimal number of operations (deletion, insertion, substitution at word level) needed to transform one path (p_1) into the other (p_2) [40].

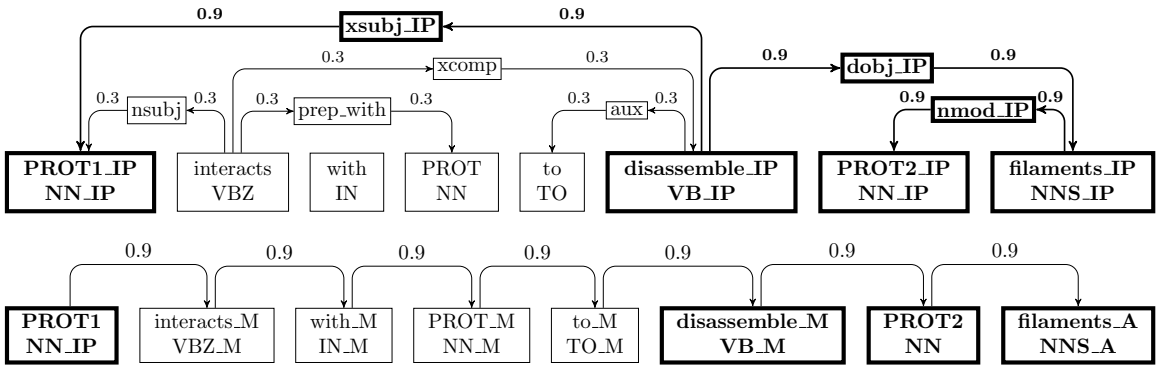


Figure 5.10: All-path graph representation. Reprinted from Airola et al. [3].

Following [40], this number is normalized by the length of the longer path and converted into a similarity measure.

$$sim_e(p_1, p_2) = e^{-\gamma \text{editdist}(p_1, p_2)} \tag{5.1}$$

When comparing two shortest paths, we considered the word lemma and the edge labels. We also renamed the candidate pair in the sentence as "E1" and "E2" and the remaining proteins provided in the annotation as "EX". For example, the following are the shortest paths of Figures 5.2a, 5.4 and 5.9.

1. E1 $\leftarrow$ *arg0* $\leftarrow$ activate $\rightarrow$ *arg1* $\rightarrow$ E2
2. E1 $\leftarrow$ *arg0* $\leftarrow$ bind $\rightarrow$ *arg1* $\rightarrow$ E2
3. E1 $\leftarrow$ *arg0* $\leftarrow$ bind $\rightarrow$ *arg1* $\rightarrow$ E2

Therefore, the edit distance between paths 1 and 2 is 1 because the predicate verbs are different. The distance between paths 2 and 3 is 0. It shows the generalizability of using EDG.

All-paths graph kernel is a practical instantiation of a graph kernel framework [43]. It counts weighted shared paths of all possible lengths in a graph [3]. As shown in Figure 5.10, all-paths graph kernel uses two graph representations: (1) a dependency graph where all edges on the shortest paths between the candidate pair receive a weight of 0.9 and other edges receive a weight of 0.3; and (2) a linear graph where each word node is connected by an edge to its succeeding word node with weight 0.9. We used words (not lemmas) and edge labels to compute the all-paths graph kernel. Similar to the case with the edit distance kernel, we

Table 5.1: Statistics of the five PPI corpora.

Corpus	Sentences	Positives	Negatives
AIMed	1,955	1,000	4,834
BioInfer	1,100	2,534	7,132
HPRD50	145	163	270
IEPA	486	335	482
LLL	77	164	166

replaced the protein names in a sentence with "E1", "E2" and "EX". All-path graph uses the graph kernel defined as

$$k(G', G'') = \sum_{i=1}^{|\mathcal{L}|} \sum_{j=1}^{|\mathcal{L}|} G'_{i,j} G''_{i,j}$$

where $\mathcal{L}$ is the set of possible labels vertices can have [43]. G' and G" are two adjacency matrix corresponding to two graph representations. The full G matrices can be thought as combinations of labels from connected pairs of vertices i and j, with a value $G_{i,j}$ that represents the strength of their connection. $k(G', G'')$ thus defines a similarity of two graphs based on the length of all walks between each pair of vertices in the graph. We use the APG software [1] to train and test the kernel. The software uses Sparse Regularized Least Squares method instead of SVM.

5.3.1.2 Experimental Setup

We evaluated our method on five PPI corpora that have been used in the community: AIMed [16], BioInfer [109], HPRD50 [42], IEPA [38], and LLL [92]. These corpora have different sizes (Table 5.1) and vary slightly in their definition of PPI [110]. Tikk et al. [132] conducted a comparison of a variety of PPI extraction systems on these corpora [2] [132]. We used the same experimental setup to evaluate our methods: self-interactions were excluded from the corpora and 10-fold document-level cross-validation is used for evaluation.

[1] http://mars.cs.utu.fi/PPICorpora/GraphKernel.html

[2] http://mars.cs.utu.fi/PPICorpora

For our experiments, we used the Charniak-Johnson parser [23] and the Stanford conversion tool with "Collapsed" setting to obtain SDG [35]. The edit distance kernel was trained with LIBSVM [22]. The APG kernel was trained with APG software.

Both these kernels have several parameters, whose settings can influence the performance. In this paper, we did not perform exhaustive systematic parameter search and optimization. We believe such parameter tuning techniques might lead to further improvements.

For the edit kernel, we set γ to 4.5, which was the value used in the original application of edit kernel on these corpora [40]. We set c in SVM to 10, which was the average best value used in [132]. For the APG kernel, we used the default settings of implementation of [3] which uses a grid parameter search for each iteration of the 10-fold cross validation. The parameter search selects the best setting based on a random set of 1,000 samples from the training sets (9 folds). If there are less than 1,000 samples, the software used the whole training set. Note that the test sets (the remaining fold) were not used for the parameter tuning.

5.3.1.3 Results

Performance, as measured by precision, recall, and F-score, is shown in Table 5.2. To provide context, we also include the results published in [132], [84], and [85]. Tikk et al. [132] reports the results of the APG kernel to be a leading performer on these 5 corpora. Miwa et al. [84] is an ensemble method that combines different systems. To provide the comparison with non-kernel methods, we also include the results published in [85], which is the state-of-the-art system on the five corpora. Miwa et al. [85] develops several systems that use a rich feature vector, combining analysis from different parsers and the values obtained from multiple kernels including the APG's score. L2-SVM and SVM-CW are among the leading SVM-based systems proposed in this paper.

Although we are using the same corpora in the study of [132], and the same implementation of the APG kernel, the results in Row 1 and Row 8 in the table are not the same. The differences are possibly due to the fact that different parsers were used and how parameters were chosen. However, we want to emphasize that all our own measurements

Table 5.2: Evaluation results of PPI detection on five corpora. Performance is reported in terms of Precision, Recall and F-score.

	Kernel	**AIMed**			**BioInfer**			**HPRD50**			**IEPA**			**LLL**		
		P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
1	Tikk et al. [132]	53.6	59.9	56.2	61.3	60.2	60.7	69.8	68.2	67.8	82.6	66.6	73.1	98.0	68.0	78.4
2	Miwa et al. [84]	68.8	55.0	60.8	71.1	65.7	68.1	76.1	68.5	70.9	78.6	67.5	71.7	86.0	77.6	80.1
3	L2-SVM [85]	–	–	63.2	–	–	66.2	–	–	67.2	–	–	73.0	–	–	80.3
4	SVM-CW [85]	–	–	64.0	–	–	66.7	–	–	72.7	–	–	75.2	–	–	85.9
	Edit kernel															
5	SDG	40.0	61.4	48.4	64.7	49.5	56.1	55.8	68.4	61.5	69.6	74.7	72.0	89.6	71.7	79.7
6	EDG	57.3	65.3	61.1	57.6	59.9	58.7	66.9	75.7	71.0	69.9	76.2	72.9	85.4	74.1	79.3
7	EDG (Best)	–	–	–	–	–	–	76.7	83.3	79.9	–	–	–	92.1	78.2	84.6
	All-paths graph kernel															
8	SDG	69.0	48.0	56.6	73.5	58.8	65.3	69.3	60.1	64.4	77.9	65.4	71.1	87.8	69.9	77.8
9	EDG	66.0	52.3	58.3	72.1	56.1	63.1	71.2	62.7	66.7	75.2	65.3	69.9	82.9	69.4	75.6
10	EDG (Best)	71.3	51.1	59.5	69.2	58.7	63.5	76.1	62.6	68.7	76.1	68.2	71.9	87.2	75.3	80.8

(e.g., in Rows 5-7 or Rows 8-10) are directly comparable to each other because the same parameter settings were used for each corpus.

The first part of Table 5.2 shows results using the edit distance kernel with original dependency graph (Row 5), and with the complete EDG (Row 6). We also experimented with different configurations of EDG by dropping one of the extra edge types added in EDG. The results obtained by the best configuration are reported in Row 7. On three of the corpora, the best results are obtained by using the full EDG. However, better results were obtained on HPRD50, when the *member-collection* relations were not included and on LLL, when the *is-a* relations were not included. In the next subsection, we will address why these relations were not included.

Overall, comparing Rows 5 and 6, we obtain F-score improvements using EDG over using SDG on four corpora (except LLL), with around 10% gains on AIMed and HPRD50 and a noticeable gain in the recall. For three of the corpora (AIMed, HPRD50, and IEPA), there is an increase in both precision and recall. For BioInfer, the gain in precision slightly exceeds the loss in recall whereas in LLL the gain in precision is slightly lower than the loss in recall. When Row 7 is used for comparison, we obtain an improvement in F-score for all 5 corpora with improvement in precision and recall in 4 corpora (BioInfer being the exception). We now see over 18% F-score improvement on HPRD50.

Despite weak performance of the edit kernel using the baseline SDG, the performance of this kernel with full EDG is close to or exceeds the results of the leading PPI systems using kernel methods (Rows 1 and 2) on 4 corpora and exceeds them on these 4 corpora when results of Row 5 is considered.

The second part of Table 5.2 (Rows 8–10) shows results using the APG kernel. The EDG (Best) in Row 10 is achieved on AIMed, BioInfer and LLL by dropping the *is-a* relation and on HPRD50 by not including the *member-collection* relations. We see F-score gains on 4 corpora through the use of EDG.

Comparing the results on the edit distance and APG kernels, we find that the more complex APG kernel (the best one overall in [132] study) gets generally better results than Edit kernel using the baseline SDG. However, the use of EDG not only closes the gap between

the kernels but in fact, edit kernel with EDG obtains higher F-score than APG with SDG or EDG in 4 of the 5 corpora.

Row 3 shows the results of L2-SVM on these corpora. We observe that both edit kernel and APG kernel with EDG (Best) gets improvements on two of the corpora. Row 4 shows the results of SVM modified for corpora weighting (SVM-CW). Using one of the corpora as the target corpus, SVM-CW weights the remaining corpora (called the source corpora) with "goodness" for training on the target corpus, adjusting the effect of their compatibility and incompatibility [85]. Thus, their results are not directly comparable with our results. However we obtain improvements using edit kernel with EDG (Best) on HPRD50.

5.3.1.4 Contribution of Individual Relation

Table 5.3 compares the effects of different techniques in EDG on five corpora using the edit distance kernel. We first evaluated SDG obtained from the Stanford conversion tool with "CCProcessed" setting (Row 2) for processing conjunctions, and next added only syntax based *arg0* and *arg1* (Row 3). After that, we added in succession referential links (including coreference, appositive, and *is-a*), *member-collection*, and *part-whole* detection in the EDG construction step by step (Row 4–6). Overall, using "CCProcessed" increases the F-scores on all five corpora. EDG constructed using syntax based *arg* achieves additional increases on 4 out of 5 corpora (the exception was IEPA). Every subsequent step generally provides more improvements on F-scores. However, we observed that on HPRD50, *member-collection* decreased F-score. Therefore we tried to switch off this part in the EDG construction but included the rest of the relations and achieved a higher F-score of 79.9% on this corpus (Row 7). This corresponds to the same result we displayed in Row 5 (EDG Best) in Table 5.2. On the LLL corpus, as components were successively added, we noticed a drop in F-score when referential linking was added. So similarly by turning off *is-a* detection and including all other EDG edges enabled us to obtain the EDG best F-score of 84.6% on LLL.

We also identified that *is-a* decreased F-scores on IEPA, however no further improvement could be made by switching it off.

Table 5.3: Contributions of different part in SDG and EDG using edit kernel. Performance is reported in terms of Precision, Recall and F-score.

	Kernel	**AIMed**			**BioInfer**			**HPRD50**			**IEPA**			**LLL**		
		P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
1	SDG (Collapsed)	40.0	61.4	48.4	64.7	49.5	56.1	55.8	68.4	61.5	69.6	74.7	72.0	89.6	71.7	79.7
2	SDG (CCProcessed)	46.4	58.9	51.9	56.2	57.1	56.6	58.9	67.6	63.0	70.2	74.8	72.4	89.6	73.5	80.8
3	EDG (syntax based *arg*)	48.1	61.2	53.9	56.3	58.5	57.4	66.9	73.2	69.9	69.3	74.4	71.7	89.0	74.1	80.9
4	EDG (+ *coref*, *app*, *isa*)	52.2	58.6	55.2	56.7	58.3	57.5	65.6	77.0	70.9	69.0	74.0	71.4	87.2	72.2	79.0
5	EDG (+ *mem-coll*)	53.2	59.2	56.0	57.1	58.6	57.8	64.4	77.8	70.5	69.6	76.4	72.8	85.4	74.5	79.6
6	EDG (+ *part-whole*)	57.3	65.3	61.1	57.6	59.9	58.7	66.9	75.7	71.0	69.9	76.2	72.9	85.4	74.1	79.3
7	EDG (Best)	57.3	65.3	61.1	57.6	59.9	58.7	76.7	83.3	79.9	69.9	76.2	72.9	92.1	78.2	84.6

Additionally, due to the gap in the performance between our system and [84] on BioInfer, we analyzed the error cases and noticed several cases similar to the following example. The candidate pair of named entities is marked in bold.

Example 23 This process involves other **actin-binding proteins**, such as **cofilin** and coronin.

Using techniques as shown in Figure 5.4, we can create *arg0 (binding, actin)* and *arg1 (binding, proteins)* in EDG and also detect *member-collection* relation between "actin-binding proteins" and "cofilin". With propagation, an interaction between "actin" and "cofilin" can be predicted. However, this relation is annotated as a negative, but instead the annotation in BioInfer includes a positive relation between "actin-binding proteins" and "cofilin". Because of similar examples in BioInfer, the *member-collection* and *is-a* and propagation failed to improve the results in BioInfer.

5.3.2 Evaluation of a Rule-based System

EDG is designed for the general purpose of biomedical relation extraction, not only for machine-learning methods, but also for rule-based systems. To test its later usage, we used BioNLP-ST (Shared Tasks) 2011 GE task corpus to test the utility of EDG in a rule-based system. Note that, EDG abstracts and continuely generalizes lingusitic and domain knowledge behind the framework discussed in Chapter 4 to an intermediate representation of sentences. Thus the rule-based system developed in this section can be considered as the evolution of iXtractR.

We use predicate-argument rules on EDG to extract GE pairs <Trigger, Theme>. The predicate can be "transcribe", "interact", "localize" etc. that triggers the potential occurrence of an event (e.g., Transcription, Binding, and Localization respectively). More details about trigger selection can be found in Section 4.3.2.2 Since EDG applies lemmatization to abstract from different inflectional forms of words, only the common base forms of triggers (lemma) are used in the system.

Because numbered arguments and their propagation provide a uniform representation for various textual variations, EDG allows the number of rules to extract GE event to greatly reduce. In our system, only two sets of rules are used for "Simple Event". Here we consider

arg1 because Simple Event extraction only requires primary argument, which is the theme of the event.

1. verb trigger $\xrightarrow{arg1}$ Entity
2. noun trigger $\xrightarrow{arg1}$ Entity

Rule 1 is a set of most basic and strict rules. Because EDG has unified different forms of predicates in the vertices, we only need to list stems of triggers in the rules. Rule 1 employs trigger stems that are verbal but, of course, can match noun forms such as "association" and "phosphorylation" in the text. Rule 2 accounts for triggers that are not derived from verbs (e.g., "transcript" and "proteolysis"). This rule matches the noun phrase such as "transcript of $\left[\mathrm{X}\right]_{\text{entity}}$". Moreover, to correctly differentiate Transcription with Gene_expression, we require that the entity of Transcription to be the type of "mRNA".

For Binding event, the trigger is non-directional, such as "bind" and "associate". Its two *arg*s undergo the same effect, meaning the order of *arg0* and *arg1* can be swapped. Therefore, we use the rule "verb trigger $\xrightarrow{arg0}$ Entity" as well. Further, we require *arg0* of the Binding triggers has to be shown, however it can link either to a marked entity or to a definite noun such as "this protein".

Once an EDG is produced for a sentence, the above rules are matched with the EDG using a subgraph-matching algorithm [77]. For each rule, a subgraph is constructed. Both nodes and EDGs in the subgraph are predicates of EDG nodes and EDGs. The worst-case complexity of the subgraph matching algorithm is $O(n^2k^n)$ where n is the number of vertices in EDG and k is the vertex degree. It is worth noting that we only use *arg0* and *arg1* in the rules, thus EDG only contains numbered arguments, and the matching is efficient in practice.

Table 5.4 summarizes the performance of our system on the development set and the test set of the GE corpus. The results were obtained using the online evaluation system.[3,4]

We provide results for the Simple Event averaging over five events, results for each of the six individual events including Binding, as well as the overall results for all events.

[3] http://bionlp-st.dbcls.jp/GE/2011/eval-development/

[4] http://bionlp-st.dbcls.jp/GE/2011/eval-test/

Table 5.4: Evaluation results from the development set and test sets of BioNLP-ST 2011 GE task. Performance is reported in terms of Precision, Recall and F-score.

	Devel set			Test set		
	P	**R**	**F**	**P**	**R**	**F**
iXtractR total	91.92	58.46	71.47	93.96	55.79	69.28
Simple Event	**89.24**	**68.14**	**77.28**	**89.78**	**62.80**	**73.90**
Gene_expression	89.90	70.09	78.77	90.58	64.27	75.19
Transcription	80.81	50.63	62.26	81.82	51.72	63.38
Protein_catabolism	94.44	73.91	82.93	100.00	73.33	84.62
Phosphorylation	93.26	74.77	83.00	92.00	74.59	82.39
Localization	89.29	74.63	81.30	88.60	52.88	66.23
Binding	**82.07**	**59.92**	**69.27**	**80.00**	**47.92**	**59.93**
Total	**87.12**	**65.62**	**74.86**	**87.44**	**58.80**	**70.31**

Overall, we obtained a global F-score of 74.86% on the development set and 70.31% on the test set. Both are better than the system under the same framework but without using EDG (Section 4.3.2.4). The gain are contributed by the increase of recall because EDG provide a more unified forms sentence constructs. As a result, even with a few of rules, we are able to cover a great range of text variations. On the other hand, we also observed a drop on precision. Error analysis indicates that entities with "dash", such as "IRF-4-positive cells", are not tokenized well in EDG, which leads to incorrect dependency edges from "cells" to "IRF-4".

5.4 Conclusion

In this chapter, we strive to find a level of abstraction that is more suitable for tasks such as relation extraction. For this purpose, we introduced techniques to create a new dependency graph representation (EDG) that goes beyond syntactic dependencies.

We evaluated the efficacy of EDG in both machine-learning and rule-based systems. In the machine-learning systems, we evaluated EDG with the edit distance and APG kernels and applied them on 5 different PPI-related datasets. We obtained improvements in F-score by using EDG. We find that despite the simplicity of the edit kernel and its weak performance with the baseline graph, results comparable to state-of-the-art systems using kernel methods

are obtained on different corpora with the inclusion of EDG.

In the rule-based system, we re-implemented the framework described in Chapter 4 and applied it to BioNLP 2011 GE task. We show the use of a few rules on EDG still enables us to get good coverage of GE event detection. This, in particular, allows us to address one of the main criticisms against rule-based systems – it is hard to develop rules for all the variations found in the text. We believe this information is not task-dependent and an enhanced understanding will contribute to developing systems for various relation extraction tasks, including genetic interactions.

In future, we plan to test the use of EDG on other relation extraction tasks in the biomedical domain. We also plan to investigate richer features and their combinations in conjunction with the use of EDG.

Chapter 6

CONCLUSIONS

Biomedical relation extraction plays an important role in automatically gathering facts and evidence for life sciences. Although many efforts have been proposed and studied, this task remains challenging due to the factors such as the complexity of the biomedical text, the variety of relations to extract, the availability of annotated corpora, the adaptation to new domains, and the selection of NLP techniques. As an attempt to address these issues, this thesis studied different representations of biomedical text by incorporating dependencies and other linguistic information. Our representation helps in the design of rule-based or machine-learning methods. This chapter contains a summary discussion of thesis contributions and directions for future work.

6.1 Thesis Summary and Contributions

The main contributions of this thesis are the methods that we proposed to analyze the linguistic structures of biomedical text and the enhancement to the existing relation extraction tasks in terms of performance, development time, and generalizability. More specific, our contributions through three steps are as follows:

1. In this thesis, we have presented our approach for sentence simplification [101]. Its aim is to make text easier to process by relation extraction programs. We described an automatic sentence simplification system, iSimp, which can detect various syntactic constructs that are frequently encountered in the biomedical literature, including but not limited to coordination, relative clause, and apposition. We also described rules for detecting syntactic constructs with emphasis on the boundary detection and on how it handle nested simplification constructs. We demonstrated that iSimp compares favorably with other simplifiers reported in the biomedical domain and it evaluates well on the types of constructs used in our approach.

We further enhanced iSimp to fully adopt the BioC format [102, 104]. We proposed a unique schema, which contains a BioC tag set for annotating simplification results and proposed a schema that allows simplified sentences to be included in the BioC annotation file and be treated as part of the original collection. The proposed schema is different than the standard schema in that it can include words that are not part of the original text.

To illustrate the usefulness of iSimp, we examined its impact on different NLP systems ranging from the relation and open information extraction to sentence selection. These evaluations showed that, with sentence simplification provided by iSimp, the recalls could generally increase without introducing precision errors. In addition, the study set up corpora for evaluating simplification performance in the BioC format. The corpora may be used as public benchmarking corpora.

2. We described a framework to facilitate the development of a pattern-based biomedical relation extraction system [103]. It aims to address the issue of substantial time and effort required for designing and implementing rule-based relation extraction systems. Developing rule-based systems often requires extensive effort from domain experts, who are familiar with rule engineering and also in the target domain, to write extraction rules. In contrast, our framework only requires as input a list of triggers and their specifications to retrieve relations of interest. It then can utilize linguistic generalizations to speed up the development process. In particular, it leverages syntactic variations possible in a language to automatically generate lexico-syntactic patterns, applies sentence simplification, and exploits referential relations to extent the coverage of patterns.

To evaluate the performance of the framework, we implemented two rule-based systems, one for various GENIA event extraction and one for miRNA-target relation extraction. Both systems outperformed or achieved the state-of-the-art performance by boosting the recall while preserving the high precision. The fact that only trigger specification is required from domain experts, together with the fact that no training set is required, meets our goals for developing the framework: ability to create effective relation extraction systems for new relations where resources (e.g., annotated corpus or database) are not publicly available.

3. By extending these ideas, we further developed Extended Dependency Graph

(EDG) [105, 106]. It aims to alleviate the textual variations challenge by providing a unified representation of the predicate-argument structure of various text. Through the use of numbered argument labels and detection of different sentence structures, EDG goes beyond syntactic dependencies and provides a level of abstraction that is more suitable for relation extraction tasks. We believe the semantic dependencies between entities discussed in EDG are critical for either pattern-based or machine learning systems. In rule-based systems, we can apply only a few rules on EDG and are still able to get good coverage of relation detection. In machine-learning systems, we get benefit from EDG because it helps reduce the complexity of learning methods and makes kernel methods generalize well to new domains.

We assessed the efficacy of EDG in both rule-based and machine-learning systems. We obtained improvements in F-scores by using EDG in both cases, and the results are comparable to state-of-the-art systems on different corpora with the inclusion of EDG. In particular, our rule-based extraction method is simple but generalizes well on both the abstract and full-text datasets over various relation extraction tasks. Our machine-learning system used a simple kernel but outperforms the state-of-the-art systems on cross-corpora evaluations on protein-protein interaction.

Besides the utility of EDG discussed in the thesis, EDG is being used in the development of other relation extraction systems. These include the relation between proteins and complexes they belong to, the relation between mutation and diseases, and the relation between an miRNA and processes and diseases.

6.2 Future Work

In addition to the relation extraction methods that we have proposed, our study opens up several opportunities for future work.

1. Generalize EDG and broaden the scope of its usage. In this thesis, EDG has incorporated dependencies and simple linguistic information into a unified representation. In the next step, we would also like to generalize EDG to incorporate richer information such as named entity types and normalization and linear order. Also, we will consider EDG to take simple biomedical relations as arguments rather than just entities (e.g., genes or proteins).

By this way, EDG could be used to extract higher order relations such as regulation and microRNA-disease association. This will be helpful to extend relation extraction to assist in knowledge base and ontology construction in future.

So far, we have applied EDG for the tasks of extracting protein-protein interaction, six GENIA events, and chemical-disease relations. In future, we also attempt to broaden the use of EDG to other relation extraction tasks in the biomedical domain. This direction can be further fulfilled by developing appropriate kernels that are suitable for EDG, or by designing richer-feature methods and their combinations in conjunction with the use of EDG.

Another main motivation of EDG is to develop methods to learn with small datasets. We would like to explore machine-learning methods can still generalize well by using the abstraction captured in EDG. Following this direction, we plan to investigate the testing of learning with small datasets and to use EDG in the context of active learning or unsupervised techniques.

2. Relation detection in full-text articles remains challenging. While the use of EDG and other linguistic and domain knowledge has shown promising results on abstracts, relation detection in full-text articles remains challenging, in particular, how to detect relations that are across sentences. Most of the cross-sentence relation extraction appears to be concerned with coreference. Thus one possible way is anaphora resolution, which finds all expressions that refer to the same entity in a discourse [71]. It is a foundational yet challenging natural language processing task which, if performed successfully, is likely to enhance systems significantly. Though much work has been studied for general named entities, such as person, organization, and location, few has explored the coreference of biomedical named entities [57, 60, 94]. We believe knowledge of the linguistic and biomedical domain play key roles in restricting the number of antecedents for anaphora.

EDG provides an ideal text representation for exploiting such information. Through coreference resolution, we are able to connect EDG in a discourse. As a result, EDG is able to provide both sentence-level and document-level information. For biomedical relation and event extraction, this benefit may provide a variety of cross-sentence patterns that can be

designed in the rule-based system, and non-local features that can be shared in the machine-learning system.

3. Incorporation of EDG in practical relation extraction systems. While the work in this thesis has emphasized on a new approach to relation extraction and the underlying principles, there are a few necessary steps that are needed to use EDG in the development of practical relation extraction systems. For example, in our evaluation, we have assumed that the named entities are provided with the text. But a real relation extraction system has to include named entity recognition step prior to the use of EDG. In the future, we would like to develop the entire relation extraction pipeline and evaluate it. Moreover, the joint modeling of several levels of information extraction can also be explored. Bayesian networks or joint graphic models are used to combine named entity recognition, coreference resolution and relation extraction [68, 118, 125]. Such joint modeling approached may help to avoid cascading errors, and are also interesting for further investigation.

4. Scalability and interoperability of relation extraction systems. With the rapid growth of publications, relation extraction on large-scale volumes of documents becomes more important. For example, to allow us to create a database of the protein-protein interaction results for the online search, we need to conduct full-scale PPI extraction to process all the MEDLINE abstracts and all the full-length articles in the PubMed Central Open Access Subset. Most efforts to construct EDG and extract relations discussed in this thesis have been evaluated on limited-scale corpora. Their usefulness for supporting large-scale discovery is still unknown. In the future, we plan to apply EDG on large datasets of abstracts and full-text articles and investigate techniques such as intermediate data storage and exchange, error detection, and crash recovery.

Bibliography

[1] Caroline B Ahlers, Marcelo Fiszman, Dina Demner-Fushman, François-Michel Lang, and Thomas C Rindflesch. Extracting semantic predications from MEDLINE citations for pharmacogenomics. In Proceedings of the Pacific Symposium on Biocomputing, volume 12, pages 209–20, 2006.

[2] Syed Toufeeq Ahmed, Deepthi Chidambaram, Hasan Davulcu, and Chitta Baral. Intex: a syntactic role driven protein-protein interaction extractor for bio-medical text. In Proceedings of the ACL-ISMB Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics, pages 54–61, 2005.

[3] Antti Airola, Sampo Pyysalo, Jari Björne, Tapio Pahikkala, Filip Ginter, and Tapio Salakoski. All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning. BMC Bioinformatics, 9(Suppl 11):1–12, 2008.

[4] Artemis Alexiadou, Paul Law, Andrè Meinunger, and Chris Wilder, editors. The syntax of relative clauses. John Benjamins Publishing Company, Philadelphia, PA, USA, 2000.

[5] Nguyen Bach and Sameer Badaskar. A review of relation extraction. Technical report, CMU, 2007.

[6] Shweta Bagewadi, Tamara Bobić, Martin Hofmann-Apitius, Juliane Fluck, and Roman Klinger. Detecting miRNA mentions and relations in biomedical literature. F1000Research, 3(205):1–33, oct 2015.

[7] Adam Berger, Stephen Della Pietra, and Vincent Della Pietra. A maximum entropy approach to natural language processing. Computational Linguistics, 22(1):39–71, 1996.

[8] Jari Björne and Tapio Salakoski. Generalizing biomedical event extraction. In Proceedings of the BioNLP Shared Task 2011 Workshop, BioNLP Shared Task '11, pages 183–191, Stroudsburg, PA, USA, 2011.

[9] Jari Björne, Flip Ginter, Sampo Pyysalo, Jun'ichi Tsujii, and Tapio Salakoski. Complex event extraction at PubMed scale. Bioinformatics, 26(12):i382–390, 2010.

[10] Christian Blaschke and Alfonso Valencia. The potential use of SUISEKI as a protein interaction discovery tool. Genome Informatics Series, 12:123–134, 2001.

[11] Claire Bonial, Jena Hwang, Julia Bonn, Kathryn Conger, Olga Babko-Malaya, and Martha Palmer. English PropBank annotation guidelines. Center for Computational Language and Education Research, Institute of Cognitive Science, University of Colorado at Boulder, Boulder, CO, USA, Nov 2012.

[12] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory, pages 144–152, 1992.

[13] Joan Bresnan. Lexical-functional syntax. Wiley-Blackwell, 2001.

[14] Quoc-Chinh Bui, Sophia Katrenko, and Peter M. A. Sloot. A hybrid approach to extract protein-protein interactions. Bioinformatics, 27(2):259–265, January 2011.

[15] Razvan C. Bunescu and Raymond J. Mooney. A shortest path dependency kernel for relation extraction. In Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT-EMNLP), pages 724–731, Stroudsburg, PA, USA, 2005.

[16] Razvan C. Bunescu, Ruifang Ge, Rohit J. Kate, Edward M. Marcotte, Raymond J. Mooney, Arun K. Ramani, and Yuk Wah Wong. Comparative experiments on learning information extractors for proteins and their interactions. Artificial Intelligence in Medicine, 33(2):139–155, 2005.

[17] Ekaterina Buyko, Erik Faessler, Joachim Wermter, and Udo Hahn. Syntactic simplification and semantic enrichment - Trimming dependency graphs for event extraction. Computational Intelligence, 27(4):610–644, 2011.

[18] John Carroll, Guido Minnen, Yvonne Canning, Siobhan Devlin, and John Tait. Practical simplication of English newspaper text to assist aphasic readers. In Proceedings of the AAAI98 Workshop on Integrating Artificial Intelligence and Assistive Technology, pages 7–10, 1998.

[19] Ronald Carter and Michael McCarthy. Cambridge grammar of English paperback with CD ROM: a comprehensive guide. Cambridge University Press, Cambridge, UK, 2006.

[20] Raman Chandrasekar and Bangalore Srinivas. Automatic induction of rules for text simplification. Knowledge-Based Systems, 10:183–190, 1997.

[21] Raman Chandrasekar, Christine Doran, and Bangalore Srinivas. Motivations and methods for text simplification. In Proceedings of the 16th Conference on Computational linguistics, volume 2, pages 1041–1044, Stroudsburg, PA, USA, 1996.

[22] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2(27):1–27, 2011.

[23] Eugene Charniak and Mark Johnson. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (ACL), pages 173–180, 2005.

[24] John Chen and K Vijay-Shanker. Automated extraction of TAGs from the Penn Treebank. In New Developments in Parsing Technology, volume 23, pages 73–89. Springer, 2005.

[25] Sung-Pil Choi and Sung-Hyon Myaeng. Simplicity is better: revisiting single kernel PPI extraction. In Proceedings of the 23rd International Conference on Computational Linguistics, pages 206–214, 2010.

[26] Rajesh Chowdhary, Jinfeng Zhang, and Jun S. Liu. Bayesian inference of protein-protein interactions from biological literature. Bioinformatics, 25(12):1536–1542, June 2009.

[27] Faisal Md. Chowdhury, Alberto Lavelli, and Alessandro Moschitti. A study on dependency tree kernels for automatic extraction of protein-protein interaction. In Proceedings of BioNLP 2011 Workshop, pages 124–133, Portland, OR, USA, June 2011.

[28] Hong-Woo Chun, Yoshimasa Tsuruoka, Jin-Dong Kim, Rie Shiba, Naoki Nagata, Teruyoshi Hishiki, and Juníchi Tsujii. Extraction of gene-disease relations from MEDLINE using domain dictionaries and machine learning. In Proceedings of the Pacific Symposium on Biocomputing, volume 11, pages 4–15, 2006.

[29] K Bretonnel Cohen, Karin Verspoor, Helen L Johnson, Chris Roeder, Philip V Ogren, William A Baumgartner Jr, Elizabeth White, Hannah Tipney, and Lawrence Hunter. High-precision biological event extraction: effects of system and of data. Computational Intelligence, 27(4):681–701, 2011.

[30] Michael Collins. Head-driven statistical models for natural linguage parsing. PhD thesis, University of Pennsylvania, 1999.

[31] Donald C. Comeau, Rezarta Islamaj Doğan, Paolo Ciccarese, Kevin Bretonnel Cohen, Martin Krallinger, Florian Leitner, Zhiyong Lu, Yifan Peng, Manabu Torii Fabio Rinaldi, Alfonso Valencia, Karin Verspoor, Thomas C. Wiegers, Cathy H. Wu, and W. John Wilbur. BioC: a minimalist approach to interoperability for biomedical text processing. Database, 2013:1–15, 2013.

[32] William Coster and David Kauchak. Learning to simplify sentences using wikipedia. In Proceedings of the workshop on monolingual text-to-text generation, pages 1–9. Association for Computational Linguistics, 2011.

[33] Michael A. Covington. A fundamental algorithm for dependency parsing. In Proceedings of the 39th Annual ACM Southeast Conference, pages 95–102, 2001.

[34] Nikolai Daraselia, Anton Yuryev, Sergei Egorov, Svetalana Novichkova, Alexander Nikitin, and Ilya Mazo. Extracting human protein interactions from MEDLINE using a full-sentence parser. Bioinformatics, 20(5):604–611, 2004.

[35] Marie-Catherine De Marneffe and Christopher D Manning. The Stanford typed dependencies representation. In Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation, pages 1–8, 2008.

[36] Marie-Catherine De Marneffe and Christopher D Manning. Stanford typed dependencies manual. Stanford University, apr 2015.

[37] Siobhan Devlin and John Tait. The use of a psycholinguistic database in the simplification of text for aphasic readers. Linguistic Databases, pages 161–173, 1998.

[38] Jing Ding, Daniel Berleant, Dan Nettleton, and E Wurtele. Mining MEDLINE: abstracts, sentences, or phrases. In Proceedings of the Pacific Symposium on Biocomputing, volume 7, pages 326–337, 2002.

[39] Mark Dras. Tree adjoining grammar and the reluctant paraphrasing of text. PhD thesis, Macquarie University NSW 2109 Australia, 1999.

[40] Günes Erkan, Arzucan Özgür, and Dragomir R Radev. Semi-supervised classification for extracting protein interaction sentences using dependency parsing. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), volume 7, pages 228–237, Prague, Czech Republic, 2007.

[41] Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S. Weld. Open information extraction from the web. Communications of the ACM, 51(12):68–74, 2008.

[42] Katrin Fundel, Robert Küffner, and Ralf Zimmer. RelEx-relation extraction using dependency parse trees. Bioinformatics, 23(3):365–371, 2007.

[43] Thomas Gärtner, Peter Flach, and Stefan Wrobel. On graph kernels: hardness results and efficient alternatives. In Conference on Learning Theory, pages 129–143, 2003.

[44] Matthew Gerber and Joyce Y Chai. Semantic role labeling of implicit arguments for nominal predicates. Computational Linguistics, 38(4):755–798, 2012.

[45] Claudio Giuliano, Alberto Lavelli, and Lorenza Romano. Exploiting shallow linguistic information for relation extraction from biomedical literature. In Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics, pages 3–7, April 2006.

[46] XTAG Research Group et al. A lexicalized tree adjoining grammar for English. Technical report, Technical Report IRCS-01-03, IRCS, University of Pennsylvania, 2001.

[47] Jörg Hakenberg, Robert Leaman, Nguyen Ha Vo, Siddhartha Jonnalagadda, Ryan Sullivan, Christopher Miller, Luis Tari, Chitta Baral, and Graciela Gonzalez. Efficient extraction of protein-protein interactions from full-text articles. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 7(3):481–494, 2010.

[48] Reinhard Rudolf Karl Hartmann and F. C. Stork. Dictionary of language and linguistics. John Wiley & Sons Inc, New York, 1972.

[49] Marti A. Hearst. Automatic acquisition of hyponyms from large text corpora. In Proceedings of the 15th International Conference on Computational Linguistics (COLING), pages 539–545, Stroudsburg, PA, USA, 1992.

[50] Zhang-Zhi Hu, Meenakshi Narayanaswamy, K. E. Ravikumar, K. Vijay-Shanker, and Cathy H. Wu. Literature mining and database annotation of protein phosphorylation using a rule-based system. Bioinformatics, 21(11):2759–2765, 2005.

[51] Minlie Huang, Xiaoyan Zhu, and Ming Li. A hybrid method for relation extraction from biomedical literature. International Journal of Medical Informatics, 75(6):443–455, jun 2006.

[52] Rodney D. Huddleston and Geoffrey K. Pullum. The Cambridge grammar of the English language. Cambridge University Press, Cambridge, UK, 2002.

[53] Lawrence Hunter, Zhiyong Lu, James Firby, William A Baumgartner, Helen L Johnson, Philip V Ogren, and K Bretonnel Cohen. OpenDMAP: an open source, ontology-driven concept analysis engine, with applications to capturing knowledge regarding protein transport, protein interactions and cell-type-specific gene expression. BMC Bioinformatics, 9(78):1–11, 2008.

[54] Siddhartha Jonnalagadda and Graciela Gonzalez. BioSimplify: an open source sentence simplification engine to improve recall in automatic biomedical information extraction. In Proceedings of AMIA Annual Symposium, pages 351–355, 2010.

[55] Yoshinobu Kano, Jari Björne, Filip Ginter, Tapio Salakoski, Ekaterina Buyko, Udo Hahn, K Bretonnel Cohen, Karin Verspoor, Christophe Roeder, Lawrence E Hunter, et al. U-Compare bio-event meta-service: compatible BioNLP event extraction services. BMC Bioinformatics, 12(1):481, 2011.

[56] Halil Kilicoglu and Sabine Bergler. Adapting a general semantic interpretation approach to biological event extraction. In Proceedings of the BioNLP Shared Task 2011 Workshop, pages 173–182, 2011.

[57] Halil Kilicoglu, Graciela Rosemblat, Marcelo Fiszman, and Thomas C. Rindflesch. Sortal anaphora resolution to enhance relation extraction from biomedical literature. BMC Bioinformatics, 17(163), apr 2016.

[58] Jin-Dong Kim, Tomoko Ohta, Sampo Pyysalo, Yoshinobu Kano, and Jun'ichi Tsujii. Overview of BioNLP'09 shared task on event extraction. In Proceedings of the

Workshop on Current Trends in Biomedical Natural Language Processing: Shared Task, pages 1–9, 2009.

[59] Jin-Dong Kim, Yue Wang, Toshihisa Takagi, and Akinori Yonezawa. Overview of Genia event task in BioNLP shared task 2011. In Proceedings of the BioNLP Shared Task 2011 Workshop, pages 7–15, Stroudsburg, PA, USA, 2011.

[60] Jin-Dong Kim, Ngan Nguyen, Yue Wang, Jun'ichi Tsujii, Toshihisa Takagi, and Akinori Yonezawa. The Genia event and protein coreference tasks of the BioNLP shared task 2011. BMC Bioinformatics, 13(Suppl 11):1–12, 2012.

[61] Jin-Dong Kim, Wang Yue, and Yasunori Yamamoto. The Genia event extraction shared task, 2013 edition - overview. In Proceedings of the Workshop on BioNLP Shared Task 2013, pages 20–27, 2013.

[62] Jin-Dong Kim, Jung jae Kim, Xu Han, and Dietrich Rebholz-Schuhmann. Extending the evaluation of genia event task toward knowledge base construction and comparison to gene regulation ontology task. BMC Bioinformatics, 16(Suppl 10):S3, June 2015.

[63] Jung-jae Kim and Dietrich Rebholz-Schuhmann. Improving the extraction of complex regulatory events from scientific text by using ontology-based inference. Journal of Biomedical Semantics, 2(Suppl 5):1–13, 2011.

[64] Seonho Kim, Juntae Yoon, and Jihoon Yang. Kernel approaches for genic interaction extraction. Bioinformatics, 24(1):118–26, November 2007.

[65] Seonho Kim, Juntae Yoon, Jihoon Yang, and Seog Park. Walk-weighted subsequence kernels for protein-protein interaction extraction. BMC Bioinformatics, 11(107):1–21, feb 2010.

[66] Karin Kipper, Anna Korhonen, Neville Ryant, and Martha Palmer. Extending VerbNet with novel verb classes. In Proceedings of the International Conference on Language Resources and Evaluation, volume 2006, pages 1027–1032, 2006.

[67] Peter Kolb. Experiments on the difference between semantic similarity and relatedness. In Proceedings of Nordic Conference on Computational Linguistics, volume 4, pages 81–88, 2009.

[68] Natalia Konstantinova. Review of relation extraction methods: What is new out there? In Analysis of Images, Social Networks and Texts, pages 15–28. Springer, 2014.

[69] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: probabilistic models for segmenting and labeling sequence data. In Proceedings of the International Conference on Machine Learning (ICML 01), pages 282–289, Bellevue, Washington, USA, 2001.

[70] Robert Leaman and Graciela Gonzalez. BANNER: an executable survey of advances in biomedical named entity recognition. In Pacific Symposium on Biocomputing, pages 652–663, Fairmont Orchid, Hawaii, USA, 2008.

[71] Heeyoung Lee, Angel Chang, Yves Peirsman, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. Deterministic coreference resolution based on entity-centric, precision-ranked rules. Computational Linguistics, 39(4):885–916, dec 2013.

[72] Gondy Leroy, Hsinchun Chen, and Jesse D Martinez. A shallow parser based on closed-class words to capture relations in biomedical text. Journal of Biomedical Informatics, 36(3):145–158, 2003.

[73] Beth Levin. English verb classes and alternations: a preliminary investigation. University of Chicago Press, Chicago, 1993.

[74] Beth Levin and Malka Rappaport Hovav. Argument realization. Cambridge University Press, Cambridge, UK, 2005.

[75] Roger Levy and Galen Andrew. Tregex and Tsurgeon: tools for querying and manipulating tree data structures. In Proceedings of the 5th International Conference on Language Resources and Evaluation, pages 2231–2234, 2006.

[76] Gang Li, Karen E. Ross, Cecilia N. Arighi, Yifan Peng, Cathy H. Wu, and K. Vijay-Shanker. miRTex: a text mining system for miRNA-gene relation extraction. PLoS Computational Biology, 11(9):1–24, 2015.

[77] Haibin Liu, Vlado Keselj, Christian Blouin, and Karin Verspoor. Subgraph matching-based literature mining for biomedical relations and events. In AAAI Fall Symposium Series, pages 32–37, Arlington, Virginia, USA, 2012.

[78] Haibin Liu, Karin Verspoor, Donald C Comeau, Andrew MacKinlay, and W John Wilbur. Generalizing an approximate subgraph matching-based system to extract events in molecular biology and cancer genetics. In Proceedings of the BioNLP Shared Task 2013 Workshop, pages 76–85, 2013.

[79] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. Stanford CoreNLP natural language processing toolkit. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pages 55–60, 2014.

[80] Edward M Marcotte, Ioannis Xenarios, and David Eisenberg. Mining literature for protein-protein interactions. Bioinformatics, 17(4):359–363, 2001.

[81] David McClosky. Any domain parsing: automatic domain adaptation for natural language parsing. Thesis, Department of Computer Science, Brown University, 2009.

[82] David McClosky, Mihai Surdeanu, and Christopher D Manning. Event extraction as dependency parsing for BioNLP 2011. In Proceedings of the BioNLP Shared Task 2011 Workshop, pages 41–45, 2011.

[83] John E. Miller, Manabu Torii, and K. Vijay-Shanker. Building domain-specific taggers without annotated (domain) data. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pages 1103–1111, 2007.

[84] Makoto Miwa, Rune Sætre, Yusuke Miyao, and Juníchi Tsujii. Protein-protein interaction extraction by leveraging multiple kernels and parsers. International Journal of Medical Informatics, 78(12):e39–46, 2009.

[85] Makoto Miwa, Rune Sætre, Yusuke Miyao, and Jun'ichi Tsujii. A rich feature vector for protein-protein interaction extraction from multiple corpora. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, volume 1, pages 121–130, 2009.

[86] Makoto Miwa, Rune Sætre, Yusuke Miyao, and Jun'ichi Tsujii. Entity-focused sentence simplification for relation extraction. In Proceedings of the 23rd International Conference on Computational Linguistics (COLING), pages 788–796, Stroudsburg, PA, USA, 2010.

[87] Makoto Miwa, Paul Thompson, and Sophia Ananiadou. Boosting automatic event extraction from the literature using domain adaptation and coreference resolution. Bioinformatics, 28(13):1759–1765, 2012.

[88] Makoto Miwa, Sampo Pyysalo, Tomoko Ohta, and Sophia Ananiadou. Wide coverage biomedical event extraction using multiple partially overlapping corpora. BMC Bioinformatics, 14(1):175, 2013.

[89] Raymond J Mooney and Razvan Bunescu. Mining knowledge from text using information extraction. ACM SIGKDD explorations newsletter, 7(1):3–10, 2005.

[90] Meenakshi Narayanaswamy, KE Ravikumar, and K Vijay-Shanker. A biological named entity recognizer. In Proceedings of the Pacific Symposium on Biocomputing, 8, pages 427–438, 2003.

[91] Meenakshi Narayanaswamy, K.E. Ravikumar, and K. Vijay-Shanker. Beyond the clause: extraction of phosphorylation information from MEDLINE abstracts. Bioinformatics, 21(suppl i2005):i319–i327, 2005.

[92] Claire Nédellec. Learning language in logic - genic interaction extraction challenge. In Proceedings of the 4th Learning Language in Logic Workshop, volume 7, pages 31–37, 2005.

[93] Claire Nédellec, Robert Bossy, Jin-Dong Kim, Jung jae Kim, Tomoko Ohta, Sampo Pyysalo, and Pierre Zweigenbaum. Overview of bionlp shared task 2013. In Proceedings of the BioNLP Shared Task 2013 Workshop. 2013, Association for Computational Linguistics, pages 1–7, Sofia, Bulgaria, 2013.

[94] Ngan Nguyen, Jin-Dong Kim, Makoto Miwa, Takuya Matsuzaki, and Junichi Tsujii. Improving protein coreference resolution by simple semantic classification. BMC Bioinformatics, 13(304):1–12, 2012.

[95] Nhung TH Nguyen, Makoto Miwa, Yoshimasa Tsuruoka, Takashi Chikayama, and Satoshi Tojo. Wide-coverage relation extraction from MEDLINE using deep syntax. BMC Bioinformatics, 16(107):1–11, 2015.

[96] Xia Ning and Yanjun Qi. Semi-supervised convolution graph kernels for relation extraction. In Proceedings of the Eleventh SIAM International Conference on Data Mining (SDM), pages 510–521, April 2011.

[97] Yun Niu, David Otasek, and Igor Jurisica. Evaluation of linguistic features useful in extraction of interactions from PubMed; application to annotating known, high-throughput and predicted interactions in I2D. Bioinformatics, 26(1):111–9, 2010.

[98] Stephan Oepen, Marco Kuhlmann, Yusuke Miyao, Daniel Zeman, Dan Flickinger, Jan Hajic, Angelina Ivanova, and Yi Zhang. SemEval 2014 Task 8: broad-coverage semantic dependency parsing. In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pages 63–72, 2014.

[99] Ethel Ong, Jerwin Damay, Gerard Lojico, Kimberly Lu, and Dex Tarantan. Simplifying text in medical literature. Journal of Research in Science, Computing and Engineering, 4(1):37–47, 2007.

[100] Toshihide Ono, Haretsugu Hishigaki, Akira Tanigami, and Toshihisa Takagi. Automated extraction of information on protein-protein interactions from the biological literature. Bioinformatics, 17(2):155–161, 2001.

[101] Yifan Peng, Catalina O. Tudor, Manabu Torii, Cathy H. Wu, and K. Vijay-Shanker. iSimp: a sentence simplification system for biomedical text. In IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pages 211–216, Philadelphia, PA, USA, Oct 2012.

[102] Yifan Peng, Catalina O. Tudor, Manabu Torii, Cathy H. Wu, and K. Vijay-Shanker. Enhancing the interoperability of iSimp by using the BioC format. In Proceedings of the BioCreative IV Challenge Evaluation Workshop, pages 5–9, 2013.

[103] Yifan Peng, Manabu Torii, Cathy H. Wu, and K. Vijay-Shanker. A generalizable NLP framework for fast development of pattern-based biomedical relation extraction systems. BMC Bioinformatics, 15(285):1–18, 2014.

[104] Yifan Peng, Catalina O. Tudor, Manabu Torii, Cathy H. Wu, and K. Vijay-Shanker. iSimp in BioC standard format: enhancing the interoperability of a sentence simplification system. Database, 2014(bau038):1–8, may 2014.

[105] Yifan Peng, Samir Gupta, Cathy H Wu, and K Vijay-Shanker. An extended dependency graph for relation extraction in biomedical texts. In Proceedings of BioNLP 2015 Workshop, pages 21–30, Beijing, China, 2015.

[106] Yifan Peng, Cecilia Arighi, Cathy H. Wu, and K. Vijay-Shanker. BioC-compatible full-text passage detection for protein-protein interactions using extended dependency graph. Database, 2016:baw072, 2016.

[107] Tu Minh Phuong, Doheon Lee, and Kwang Hyung Lee. Learning rules to extract protein interactions from biomedical text. In Advances in Knowledge Discovery and Data Mining, pages 148–158. Springer, 2003.

[108] Carl Pollard and Ivan A Sag. Head-driven phrase structure grammar. University of Chicago Press, Chicago, 1994.

[109] Sampo Pyysalo, Filip Ginter, Juho Heimonen, Jari Björne, Jorma Boberg, Jouni Järvinen, and Tapio Salakoski. BioInfer: a corpus for information extraction in the biomedical domain. BMC Bioinformatics, 8(50):1–24, feb 2007.

[110] Sampo Pyysalo, Antti Airola, Juho Heimonen, Jari Björne, Filip Ginter, and Tapio Salakoski. Comparative analysis of five protein-protein interaction corpora. BMC Bioinformatics, 9(Suppl 3):1–11, 2008.

[111] Longhua Qian, Guodong Zhou, Fang Kong, Qiaoming Zhu, and Peide Qian. Exploiting constituent dependencies for tree kernel-based semantic relation extraction. In Proceedings of the 22nd International Conference on Computational Linguistics, pages 697–704, Stroudsburg, PA, USA, 2008.

[112] Long Qiu, Min yen Kan, and Tat seng Chua. A public reference implementation of the rap anaphora resolution algorithm. In Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC), pages 291–294, 2004.

[113] Sebastian Riedel and Andrew McCallum. Robust biomedical event extraction with dual decomposition and minimal domain adaptation. In Proceedings of the BioNLP Shared Task 2011 Workshop, pages 46–50, 2011.

[114] Sebastian Riedel, David McClosky, Mihai Surdeanu, Andrew McCallum, and Christopher D. Manning. Model combination for event extraction in BioNLP 2011. In Proceedings of the BioNLP Shared Task 2011 Workshop, BioNLP Shared Task '11, pages 51–55, Stroudsburg, PA, USA, 2011.

[115] Fabio Rinaldi, Gerold Schneider, Kaarel Kaljurand, Michael Hess, and Martin Romacker. An environment for relation mining over richly annotated corpora: the case of GENIA. BMC Bioinformatics, 7(Suppl 3):1–9, nov 2006.

[116] Fabio Rinaldi, Gerold Schneider, Kaarel Kaljurand, Michael Hess, Christos Andronis, Ourania Konstandi, and Andreas Persidis. Mining of relations between proteins over biomedical scientific literature using a deep-linguistic approach. Artificial Intelligence in Medicine, 39(2):127–136, 2007.

[117] Barbara Rosario and Marti A Hearst. Multi-way relation classification: application to protein-protein interactions. In Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT-EMNLP), pages 732–739, 2005.

[118] Dan Roth and Wen-tau Yih. Global inference for entity and relation identification via a linear programming formulation. Introduction to statistical relational learning, pages 553–580, 2007.

[119] Yves Schabes. Stochastic Lexicalized Tree-adjoining Grammars. In Proceedings of the 15th International Conference on Computational Linguistics (COLING), volume 2, pages 425–432, Stroudsburg, PA, USA, 1992.

[120] Ariel S. Schwartz and Marti A. Hearst. A simple algorithm for identifying abbreviation definitions in biomedical text. In Proceedings of the Pacific Symposium on Biocomputing, volume 8, pages 451–462, 2003.

[121] Advaith Siddharthan. Syntactic simplification and text cohesion. PhD thesis, University of Cambridge, 2003.

[122] Advaith Siddharthan. Syntactic simplification and text cohesion. Research on Language and Computation, 4(1):77–109, mar 2006.

[123] Advaith Siddharthan. A survey of research on text simplification. International Journal of Applied Linguistics, 165(2):259–298, jan 2014.

[124] Advaith Siddharthan, Ani Nenkova, and Kathleen McKeown. Information status distinctions and referring expressions: An empirical study of references to people in news summaries. Computational Linguistics, 37(4):811–842, 2011.

[125] Sameer Singh, Sebastian Riedel, Brian Martin, Jiaping Zheng, and Andrew McCallum. Joint inference of entities, relations, and coreference. In Proceedings of the 2013 workshop on Automated knowledge base construction, pages 1–6. ACM, 2013.

[126] Noah A Smith. Ellipsis happens, and deletion is how. University of Maryland Working Papers in Linguistics, 11:176–191, 2001.

[127] Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. Learning syntactic patterns for automatic hypernym discovery. In Advances in Neural Information Processing Systems 17, pages 1297–1304, Cambridge, MA, USA, 2005. MIT Press.

[128] Lucia Specia, Sujay Kumar Jauhar, and Rada Mihalcea. Semeval-2012 task 1: English lexical simplification. In Proceedings of the First Joint Conference on Lexical and Computational Semantics, pages 347–355. Association for Computational Linguistics, 2012.

[129] Pontus Stenetorp, Goran Topić, Sampo Pyysalo, Tomoko Ohta, Jin-Dong Kim, and Jun'ichi Tsujii. BioNLP shared task 2011: supporting resources. In Proceedings of the Workshop on BioNLP Shared Task 2011, pages 112–120, June 2011.

[130] Yuka Tateisi, Akane Yakushiji, Tomoko Ohta, and Junâichi Tsujii. Syntax annotation for the GENIA corpus. In Proceedings of the Workshop on the 1st International Joint Conference on Natural Language Processing, volume 5, pages 222–227, Jeju Island, Korea, 2005.

[131] James Thomas, David Milward, Christos Ouzounis, Stephen Pulman, and Mark Carroll. Automatic extraction of protein interactions from scientific. In Proceedings of the Pacific Symposium on Biocomputing, volume 5, pages 538–549, 2000.

[132] Domonkos Tikk, Philippe Thomas, Peter Palaga, Jörg Hakenberg, and Ulf Leser. A comprehensive benchmark of kernel methods to extract protein-protein interactions from literature. PLoS Computational Biology, 6(7):1–19, jul 2010.

[133] Catalina O. Tudor and K. Vijay-Shanker. $\text{Rank}_{\text{pref}}$: ranking sentences describing relation between biomedical entities with an application. In Proceedings of BioNLP 2012 Workshop, pages 163–171, Stroudsburg, PA, USA, 2012.

[134] Sofie Van Landeghem, Sampo Pyysalo, Tomoko Ohta, and Yves Van de Peer. Integration of static relations to enhance event extraction from text. In Proceedings of the 2010 Workshop on Biomedical Natural Language Processing, pages 144–152, Stroudsburg, PA, USA, 2010.

[135] Sofie Van Landeghem, Jari Björne, Thomas Abeel, Bernard De Baets, Tapio Salakoski, and Yves Van de Peer. Semantically linking molecular entities in literature through entity relationships. BMC Bioinformatics, 13(Suppl 11):1–9, 2012.

[136] Sofie Van Landeghem, Jari Björne, Chih-Hsuan Wei, Kai Hakala, Sampo Pyysalo, Sophia Ananiadou, Hung-Yu Kao, Zhiyong Lu, Tapio Salakoski, Yves Van de Peer, et al. Large-scale event extraction from literature with multi-level gene normalization. PLoS ONE, 8(4):1–12, apr 2013.

[137] Vladimir N. Vapnik. The nature of statistical learning theory. Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[138] David Vickrey and Daphne Koller. Sentence simplification for semantic role labeling. In Proceedings of the annual meeting of the Association for Computational Linguistics, pages 344–352, 2008.

[139] Andreas Vlachos and Mark Craven. Biomedical event extraction from abstracts and full papers using search-based structured prediction. BMC Bioinformatics, 13(Suppl 11):1–11, 2012.

[140] Kristian Woodsend and Mirella Lapata. Learning to simplify sentences with quasi-synchronous grammar and integer programming. In Proceedings of the conference on empirical methods in natural language processing (EMNLP), pages 409–420. Association for Computational Linguistics, 2011.

[141] Sander Wubben, Antal Van Den Bosch, and Emiel Krahmer. Sentence simplification by monolingual machine translation. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL), pages 1015–1024. Association for Computational Linguistics, 2012.

[142] Akane Yakushiji, Yusuke Miyao, Yuka Tateisi, and Junichi Tsujii. Biomedical information extraction with predicate-argument structure patterns. In Proceedings of the 1st International Symposium on Semantic Mining in Biomedicine, pages 1–10, Hinxton, Cambridgeshire, UK, April 2005.

[143] Min Zhang, Jie Zhang, Jian Su, and Guodong Zhou. A composite kernel to extract relations between entities with both flat and structured features. In Proceedings of the joint conference of the International Committee on Computational Linguistics and the Association for Computational Linguistics (COLING-ACL), pages 825–832, Stroudsburg, PA, USA, 2006.

[144] Yijia Zhang, Hongfei Lin, Zhihao Yang, Jian Wang, and Yanpeng Li. Hash subgraph pairwise kernel for protein-protein interaction extraction. IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB), 9(4):1190–1202, 2012.

[145] Guodong Zhou, Min Zhang, Dong Hong, and Ji Qiaoming Zhu. Tree kernel-based relation extraction with context-sensitive structured parse tree information. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pages 728–736, Prague, Czech Republic, 2007.

Appendix

REPRINT PERMISSION LETTERS

OXFORD UNIVERSITY PRESS LICENSE TERMS AND CONDITIONS

Jun 16, 2016

This Agreement between Yifan Peng ("You") and Oxford University Press ("Oxford University Press") consists of your license details and the terms and conditions provided by Oxford University Press and Copyright Clearance Center.

License Number	3890900809991
License date	Jun 16, 2016
Licensed Content Publisher	Oxford University Press
Licensed Content Publication	Database
Licensed Content Title	iSimp in BioC standard format: enhancing the interoperability of a sentence simplification system:
Licensed Content Author	Yifan Peng, Catalina O. Tudor, Manabu Torii, Cathy H. Wu, K. Vijay-Shanker
Licensed Content Date	01/01/2014
Type of Use	Thesis/Dissertation
Institution name	
Title of your work	A study of relation extraction for biomedical text
Publisher of your work	n/a
Expected publication date	Jun 2016
Permissions cost	0.00 USD
Value added tax	0.00 USD
Total	**0.00 USD**
Requestor Location	Yifan Peng 76 Munro Road NEWARK, DE 19711 United States Attn: Yifan Peng
Publisher Tax ID	GB125506730
Billing Type	Invoice
Billing Address	Yifan Peng

76 Munro Road

NEWARK, DE 19711
United States
Attn: Yifan Peng

Total **0.00 USD**

Terms and Conditions

STANDARD TERMS AND CONDITIONS FOR REPRODUCTION OF MATERIAL FROM AN OXFORD UNIVERSITY PRESS JOURNAL

1. Use of the material is restricted to the type of use specified in your order details.
2. This permission covers the use of the material in the English language in the following territory: world. If you have requested additional permission to translate this material, the terms and conditions of this reuse will be set out in clause 12.
3. This permission is limited to the particular use authorized in (1) above and does not allow you to sanction its use elsewhere in any other format other than specified above, nor does it apply to quotations, images, artistic works etc that have been reproduced from other sources which may be part of the material to be used.
4. No alteration, omission or addition is made to the material without our written consent. Permission must be re-cleared with Oxford University Press if/when you decide to reprint.
5. The following credit line appears wherever the material is used: author, title, journal, year, volume, issue number, pagination, by permission of Oxford University Press or the sponsoring society if the journal is a society journal. Where a journal is being published on behalf of a learned society, the details of that society must be included in the credit line.
6. For the reproduction of a full article from an Oxford University Press journal for whatever purpose, the corresponding author of the material concerned should be informed of the proposed use. Contact details for the corresponding authors of all Oxford University Press journal contact can be found alongside either the abstract or full text of the article concerned, accessible from www.oxfordjournals.org Should there be a problem clearing these rights, please contact journals.permissions@oup.com
7. If the credit line or acknowledgement in our publication indicates that any of the figures, images or photos was reproduced, drawn or modified from an earlier source it will be necessary for you to clear this permission with the original publisher as well. If this permission has not been obtained, please note that this material cannot be included in your publication/photocopies.
8. While you may exercise the rights licensed immediately upon issuance of the license at the end of the licensing process for the transaction, provided that you have disclosed complete and accurate details of your proposed use, no license is finally effective unless and until full payment is received from you (either by Oxford University Press or by Copyright Clearance Center (CCC)) as provided in CCC's Billing and Payment terms and conditions. If full payment is not received on a timely basis, then any license preliminarily granted shall be deemed automatically revoked and shall be void as if never granted. Further, in the event that you breach any of these terms and conditions or any of CCC's Billing and Payment terms and conditions, the license is automatically revoked and shall be void as if never granted. Use of materials as described in a revoked license, as well as any use of the materials beyond the scope of an unrevoked license, may constitute copyright infringement and Oxford University

Press reserves the right to take any and all action to protect its copyright in the materials.
9. This license is personal to you and may not be sublicensed, assigned or transferred by you to any other person without Oxford University Press's written permission.
10. Oxford University Press reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this licensing transaction, (ii) these terms and conditions and (iii) CCC's Billing and Payment terms and conditions.
11. You hereby indemnify and agree to hold harmless Oxford University Press and CCC, and their respective officers, directors, employs and agents, from and against any and all claims arising out of your use of the licensed material other than as specifically authorized pursuant to this license.
12. Other Terms and Conditions:
v1.4

Questions? customercare@copyright.com or +1-855-239-3415 (toll free in the US) or +1-978-646-2777.

Title:	iSimp: A sentence simplification system for biomedicail text
Conference Proceedings:	Bioinformatics and Biomedicine (BIBM), 2012 IEEE International Conference on
Author:	Yifan Peng; Catalina O. Tudor; Manabu Torii; Cathy H. Wu; K. Vijay-Shanker
Publisher:	IEEE
Date:	4-7 Oct. 2012

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

OXFORD UNIVERSITY PRESS LICENSE TERMS AND CONDITIONS

Jun 16, 2016

This Agreement between Yifan Peng ("You") and Oxford University Press ("Oxford University Press") consists of your license details and the terms and conditions provided by Oxford University Press and Copyright Clearance Center.

License Number	3890900317264
License date	Jun 16, 2016
Licensed Content Publisher	Oxford University Press
Licensed Content Publication	Database
Licensed Content Title	BioC-compatible full-text passage detection for protein–protein interactions using extended dependency graph:
Licensed Content Author	Yifan Peng, Cecilia Arighi, Cathy H. Wu, K. Vijay-Shanker
Licensed Content Date	01/01/2016
Type of Use	Thesis/Dissertation
Institution name	
Title of your work	A study of relation extraction for biomedical text
Publisher of your work	n/a
Expected publication date	Jun 2016
Permissions cost	0.00 USD
Value added tax	0.00 USD
Total	**0.00 USD**
Requestor Location	Yifan Peng 76 Munro Road NEWARK, DE 19711 United States Attn: Yifan Peng
Publisher Tax ID	GB125506730
Billing Type	Invoice
Billing Address	Yifan Peng

76 Munro Road

NEWARK, DE 19711
United States
Attn: Yifan Peng

Total **0.00 USD**

Terms and Conditions

STANDARD TERMS AND CONDITIONS FOR REPRODUCTION OF MATERIAL FROM AN OXFORD UNIVERSITY PRESS JOURNAL

1. Use of the material is restricted to the type of use specified in your order details.
2. This permission covers the use of the material in the English language in the following territory: world. If you have requested additional permission to translate this material, the terms and conditions of this reuse will be set out in clause 12.
3. This permission is limited to the particular use authorized in (1) above and does not allow you to sanction its use elsewhere in any other format other than specified above, nor does it apply to quotations, images, artistic works etc that have been reproduced from other sources which may be part of the material to be used.
4. No alteration, omission or addition is made to the material without our written consent. Permission must be re-cleared with Oxford University Press if/when you decide to reprint.
5. The following credit line appears wherever the material is used: author, title, journal, year, volume, issue number, pagination, by permission of Oxford University Press or the sponsoring society if the journal is a society journal. Where a journal is being published on behalf of a learned society, the details of that society must be included in the credit line.
6. For the reproduction of a full article from an Oxford University Press journal for whatever purpose, the corresponding author of the material concerned should be informed of the proposed use. Contact details for the corresponding authors of all Oxford University Press journal contact can be found alongside either the abstract or full text of the article concerned, accessible from www.oxfordjournals.org Should there be a problem clearing these rights, please contact journals.permissions@oup.com
7. If the credit line or acknowledgement in our publication indicates that any of the figures, images or photos was reproduced, drawn or modified from an earlier source it will be necessary for you to clear this permission with the original publisher as well. If this permission has not been obtained, please note that this material cannot be included in your publication/photocopies.
8. While you may exercise the rights licensed immediately upon issuance of the license at the end of the licensing process for the transaction, provided that you have disclosed complete and accurate details of your proposed use, no license is finally effective unless and until full payment is received from you (either by Oxford University Press or by Copyright Clearance Center (CCC)) as provided in CCC's Billing and Payment terms and conditions. If full payment is not received on a timely basis, then any license preliminarily granted shall be deemed automatically revoked and shall be void as if never granted. Further, in the event that you breach any of these terms and conditions or any of CCC's Billing and Payment terms and conditions, the license is automatically revoked and shall be void as if never granted. Use of materials as described in a revoked license, as well as any use of the materials beyond the scope of an unrevoked license, may constitute copyright infringement and Oxford University

Press reserves the right to take any and all action to protect its copyright in the materials.
9. This license is personal to you and may not be sublicensed, assigned or transferred by you to any other person without Oxford University Press's written permission.
10. Oxford University Press reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this licensing transaction, (ii) these terms and conditions and (iii) CCC's Billing and Payment terms and conditions.
11. You hereby indemnify and agree to hold harmless Oxford University Press and CCC, and their respective officers, directors, employs and agents, from and against any and all claims arising out of your use of the licensed material other than as specifically authorized pursuant to this license.
12. Other Terms and Conditions:
v1.4

Questions? customercare@copyright.com or +1-855-239-3415 (toll free in the US) or +1-978-646-2777.

OXFORD UNIVERSITY PRESS LICENSE TERMS AND CONDITIONS

Jun 16, 2016

This Agreement between Yifan Peng ("You") and Oxford University Press ("Oxford University Press") consists of your license details and the terms and conditions provided by Oxford University Press and Copyright Clearance Center.

License Number	3890900984788
License date	Jun 16, 2016
Licensed Content Publisher	Oxford University Press
Licensed Content Publication	Database
Licensed Content Title	BioC: a minimalist approach to interoperability for biomedical text processing:
Licensed Content Author	Donald C. Comeau, Rezarta Islamaj Doğan, Paolo Ciccarese, Kevin Bretonnel Cohen, Martin Krallinger, Florian Leitner, Zhiyong Lu, Yifan Peng, Fabio Rinaldi, Manabu Torii, Alfonso Valencia, Karin Verspoor, Thomas C. Wiegers, Cathy H. Wu, W. John Wilbur
Licensed Content Date	01/01/2013
Type of Use	Thesis/Dissertation
Institution name	
Title of your work	A study of relation extraction for biomedical text
Publisher of your work	n/a
Expected publication date	Jun 2016
Permissions cost	0.00 USD
Value added tax	0.00 USD
Total	**0.00 USD**
Requestor Location	Yifan Peng 76 Munro Road NEWARK, DE 19711 United States Attn: Yifan Peng
Publisher Tax ID	GB125506730

Billing Type	Invoice
Billing Address	Yifan Peng 76 Munro Road NEWARK, DE 19711 United States Attn: Yifan Peng
Total	**0.00 USD**

Terms and Conditions

STANDARD TERMS AND CONDITIONS FOR REPRODUCTION OF MATERIAL FROM AN OXFORD UNIVERSITY PRESS JOURNAL

1. Use of the material is restricted to the type of use specified in your order details.
2. This permission covers the use of the material in the English language in the following territory: world. If you have requested additional permission to translate this material, the terms and conditions of this reuse will be set out in clause 12.
3. This permission is limited to the particular use authorized in (1) above and does not allow you to sanction its use elsewhere in any other format other than specified above, nor does it apply to quotations, images, artistic works etc that have been reproduced from other sources which may be part of the material to be used.
4. No alteration, omission or addition is made to the material without our written consent. Permission must be re-cleared with Oxford University Press if/when you decide to reprint.
5. The following credit line appears wherever the material is used: author, title, journal, year, volume, issue number, pagination, by permission of Oxford University Press or the sponsoring society if the journal is a society journal. Where a journal is being published on behalf of a learned society, the details of that society must be included in the credit line.
6. For the reproduction of a full article from an Oxford University Press journal for whatever purpose, the corresponding author of the material concerned should be informed of the proposed use. Contact details for the corresponding authors of all Oxford University Press journal contact can be found alongside either the abstract or full text of the article concerned, accessible from www.oxfordjournals.org Should there be a problem clearing these rights, please contact journals.permissions@oup.com
7. If the credit line or acknowledgement in our publication indicates that any of the figures, images or photos was reproduced, drawn or modified from an earlier source it will be necessary for you to clear this permission with the original publisher as well. If this permission has not been obtained, please note that this material cannot be included in your publication/photocopies.
8. While you may exercise the rights licensed immediately upon issuance of the license at the end of the licensing process for the transaction, provided that you have disclosed complete and accurate details of your proposed use, no license is finally effective unless and until full payment is received from you (either by Oxford University Press or by Copyright Clearance Center (CCC)) as provided in CCC's Billing and Payment terms and conditions. If full payment is not received on a timely basis, then any license preliminarily granted shall be deemed automatically revoked and shall be void as if never granted. Further, in the event that you breach any of these terms and conditions or any of CCC's Billing and Payment terms and

conditions, the license is automatically revoked and shall be void as if never granted. Use of materials as described in a revoked license, as well as any use of the materials beyond the scope of an unrevoked license, may constitute copyright infringement and Oxford University Press reserves the right to take any and all action to protect its copyright in the materials.
9. This license is personal to you and may not be sublicensed, assigned or transferred by you to any other person without Oxford University Press's written permission.
10. Oxford University Press reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this licensing transaction, (ii) these terms and conditions and (iii) CCC's Billing and Payment terms and conditions.
11. You hereby indemnify and agree to hold harmless Oxford University Press and CCC, and their respective officers, directors, employs and agents, from and against any and all claims arising out of your use of the licensed material other than as specifically authorized pursuant to this license.
12. Other Terms and Conditions:
v1.4