Title	深い統語構造を利用した生命科学文献からの関係抽出
Author(s)	Nguyen，Thi Hong Nhung
Citation	
Issue Date	2014-12
Type	Thesis or Dissertation
Text version	ETD
URL	http://hdl.handle.net/10119/12619
Rights	
Description	Supervisor:東条　敏，情報科学研究科，博士

Doctoral Dissertation

Unsupervised Relation Extraction from Biomedical Literature Using Deep Syntax

NGUYEN, Nhung Thi Hong

Supervisor: Professor Satoshi Tojo

School of Information Science
Japan Advanced Institute of Science and Technology

December 2014

Abstract

The explosive growth of published biomedical research provides scientists with the chance to find correlations or associations between biomedical concepts from the literature. Therefore, there is a growing demand to convert information in free text into more structural forms. This demand motivates many researchers and scientists to work on *relation extraction*, an information extraction task that aims to extract semantic relations between important biomedical concepts. However, most of the previous studies on this topic have focused on specific or predefined types of relations, which inherently limits the types of the extracted relations. To overcome this limitation, we propose a relation extraction system that attempts to locate all possible relations present in the input documents.

In building such a general relation extraction system, we face two challenges: (1) there is no available tool that is trained on a gold standard corpus to recognize all named entities while dictionary-based tools may generate many false positives; and (2) there is no available annotated corpus for such a general schema of relations that can be used for training the extraction model. Our proposed system relies on a dictionary-based named entity recognizer and performs some post-processing to discard spurious entities. It then deals with the second challenge by employing predicate-argument structure (PAS) patterns, which are well-normalized forms that represent deep syntactic relations. In this dissertation, we introduce six PAS patterns for binary relations. After matching the patterns to extract candidates of relations, the system checks the semantic types according to a semantic network to find true relations. Our manual evaluation on a set of 500 sentences randomly selected from MEDLINE has shown a reasonable level of performance of the system (a pseudo F-score of 55.89% on average) compared with other state-of-the-art systems, including ReVerb, OLLIE and SemRep. Our system can detect broader types of relations but less precisely than SemRep, a rule-based semantic interpreter for biomedical text. The evaluation in another setting on pre-defined relations has also shown its wider coverage.

We then have applied our system to the entire MEDLINE corpus and produced more than 137 million semantic relations. The extraction results are useful in their own right, but they also provide us with a quantitative understanding of what kinds of semantic relations are actually described in MEDLINE and can be ultimately extracted by (possibly type-specific) relation extraction systems. The entire collection of the extracted relations is publicly available in machine-readable form, so that it can serve as a potential knowledge base for high-level text-mining applications in the biomedical domain.

When using the extracted relations as an underlying database, the text-mining applications would meet the problem of spurious mismatches caused by the diversity of natural language expressions. Therefore, the second task in our dissertation is to detect synonymy between relational phrases that represent the relations to alleviate the problem of mismatches. Most of the previous work that has addressed this task uses similarity metrics between relational phrases based on textual strings or dependency paths, which, for the most part, ignore the context around the relations. To overcome this shortcoming, we employ a word embedding technique to encode relational phrases. We then apply the k-means algorithm on top of the distributional representations to cluster the phrases. Our experimental results show that this approach outperforms state-of-the-art statistical models including Latent Dirichlet Allocation and Markov Logic Networks.

Keywords: relation extraction, open information extraction, predicate-argument structure, relational phrase clustering, word embeddings.

Acknowledgements

This dissertation would not have been completed without the help, support, guidance and effort of a lot of people. It gives me great pleasure to express my sincere thanks to whom I am greatly indebted.

First of all, I owe my deepest gratitude to my supervisor, Associate Professor Yoshimasa Tsuruoka (The University of Tokyo), for his guidance and support from the first day when I am a fresh PhD student to the present day when I am going to graduate. His constant advising and encouragement have guided me through my most difficult time in research. I have also learned from him much valuable knowledge and experience in the academic life.

I am really grateful to Associate Professor Makoto Miwa (Toyota Technological Institute) for his dedicated guidance, fruitful comments and suggestions on my topic.

I would like to show my sincere thanks to Professor Satoshi Tojo and Associate Professor Thuc Dinh Nguyen (University of Science, HCMC, Vietnam), who have supported me a lot during my doctoral study. I would like to thank Associate Professor Minh Le Nguyen and Professor Hiroyuki Iida for their concern and encouragement after my preliminary defense.

I wish to express the appreciation to Professor Takashi Chikayama (The University of Tokyo), who gave me supportive comments when I did my off-campus research at The University of Tokyo.

A very special thank to the members of Tsuruoka laboratory (The University of Tokyo) and Tojo laboratory, who shared their research ideas, useful experiences and discussions, and to my trustworthy friends, who have stood by and helped me in my daily life.

I would like to thank the FIVE-JAIST project between JAIST and University of Science, HCMC, Vietnam for its financial support during my study.

Last but not least, many thanks go to my family, especially my grandmothers, my mother and sisters, who always encourage, support and believe in me throughout my life.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, we first introduce the context of our research, which relates to biomedical text mining–a general context of relation extraction. We then describe our research objectives: detecting all possible relations between two biomedical concepts in a document, and synonym resolution of relational phrases based on the output of our system on MEDLINE. Finally, we briefly present the outline of this dissertation.

1.1 Relation Extraction from Biomedical Literature

Biomedical text mining, also known as biomedical natural language processing (BioNLP), is a field of natural language processing that aims to develop methods for extracting useful information from the literature in biomedical and biology domains. The goal of biomedical text mining [26] is to provide researchers with efficient tools to search and visualize necessary information, and uncover relationships between biomedical concepts described in the vast amount of the documents. Not surprisingly, many biomedical text mining systems have been developed, such as Chilibot [20], MEDIE [89, 101], EBIMed [111], Kleio [99], PathText [58, 86], FACTA+ [144], BioCaster [28], and semantic MEDLINE [120].

Generally, a text mining task is divisible into six main subtasks including (1) sentence segmentation, (2) tokenization, (3) part-of-speech tagging, (4) parsing, (5) named entity recognition (NER), and (6) relation extraction. The final step of this pipeline–relation extraction involves using the representation of the structure of the input produced in the preceding steps to identify relations of the marked entities and recognize the types of these relations. Most of the previous work on relation extraction from biomedical literature focuses on specific or predefined types of relations, such as protein-protein interaction [155, 1, 87], gene-protein relationship [20, 101], drug-drug interaction [127], and biomolecular events [98]. The types of relations that can be extracted by existing approaches are, therefore, inherently limited.

Recently, an information extraction paradigm called Open Information Extraction (OIE) has been introduced to overcome the above-mentioned limitation [8]. OIE systems (e.g., ReVerb [35] and OLLIE [73]) aim to extract all possible relations from Web text. Although the concept of OIE is certainly appealing, our preliminary experiments have suggested that these state-of-the-art systems do not perform well on biomedical text. This is partly because biomedical literature usually uses sentences with more complex structures and specialized vocabulary compared with Web text.

Altman et al. [4] suggested that biomedical text mining should address broader types of entities beyond genes and proteins, and broader types of relations, including complex relations. This suggestion and the observation that OIE systems do not perform well on biomedical text have motivated us to develop a relation extraction that attempts to locate all possible relations present in the literature.

We have built a system that uses Predicate-Argument Structure (PAS) patterns to detect the candidates of possible biomedical relations. We decided to use PAS patterns because they are well-normalized forms that represent deep syntactic relations. In other words, multiple syntactic variations are reduced to a single PAS, thereby allowing us to cover many kinds of expressions with a small number of PAS patterns. The system first applies a deep syntactic parser to input sentences, and then matches its output to predefined PAS patterns to detect relevant noun phrases (NPs). Named entities in these NPs are then detected; and finally, relations between these entities are extracted. The output of our system is a set of all semantic relations contained in the input.

There are two challenges for our system. Firstly, it has to recognize all types of named entities, not only focus on specific ones. This NER task is really difficult since there is no available annotated corpus for all kinds of entities. Therefore, we have to employ a dictionary-based NER tool [6] in this step, which might generate many false positive entities and affect the performance of the next step. The second challenge is that unlike most of previous systems that perform their extraction models based on a specific ontology [28] or a pre-defined structure of relations with gold standard corpora [1, 87, 98, 127], our system solely relies on the textual content of documents to locate relations. This also explains for the reason why we use PAS patterns.

Perhaps the most similar and relevant to our work is SemRep [119, 120] and the system by Nebot and Berlanga [97] performed a similar task to our work. SemRep is a rule-based semantic interpreter that extracts semantic relationships from free text. Their relationships are represented as *predications*, a formal representation consisting of a predicate and arguments. SemRep extracts 30 predicate types, mostly related to clinical medicine, substance interactions, genetic etiology of disease and pharmacogenomics. Their predicates were created by modifying 30 relation types of the UMLS Semantic Network[1]. The system by Nebot and Berlanga [97] extracts explicit binary relations of the form <*subject, predicate, object*> from CALBC initiative [110]. To detect candidate relations, they proposed seven simple lexico-syntactic patterns. These patterns are expressed in part-of-speech tags in which relational phrases reside between the two entities.

We have designed our system with a particular focus on recall, in regard to its extraction performance. This is primarily because we wanted to extract all binary relations between important biomedical concepts described in the whole MEDLINE. The use of PAS patterns helped us to achieve relatively high recall (while keeping reasonable precision), because PAS patterns effectively represent many lexico-syntactic patterns at an abstract level and thus are robust to various syntactic transformations such as passivization, control constructions, relative clauses, and their combinations, which are quite common in sentences expressing biomedical relations.

We have then applied our system to the entire MEDLINE corpus, and produced more than 137 million semantic relations. The extraction results are useful in their own right, but they also provide us with a quantitative understanding of what kinds of semantic relations are actually described in MEDLINE and can be ultimately extracted by (possibly type-specific) relation extraction systems. The entire collection of the relations extracted from MEDLINE can serve as a potential knowledge base for high-level text-mining applications in the biomedical domain.

1.2 Synonym Resolution for Relational Phrases

Many of the robust text mining systems in the biomedical domain allow end-users to browse and retrieve information from their databases [89, 111, 144]. Implementing such retrieval functionality is usually straight-forward if the system is only concerned with a specific type of information, such as protein-protein interaction and gene-disease association, since it is essentially a database search problem. However, the problem becomes much more difficult

[1] http://semanticnetwork.nlm.nih.gov/

when the system is designed to cover unrestricted types of relations, which requires the relation in a query to be specified using a natural language expression, such as ‘be induced by’ or ‘result in’. Such relational phrases expressed in natural language often cause spurious mismatches between the user’s query and the textual data in the underlining database. For example, given the input query “What genes are essential for cell survival?”, the system can fail to return the result <stat1, *be critical for*, cell survival> due to the string-level mismatch between *be essential for* and *be critical for*. In most situations, *be essential for* is equivalent to *be critical for*, i.e., they form a pair of synonyms, which can be used for alleviating the mismatch problem. Knowledge of synonymous phrases is therefore beneficial in many biomedical text mining applications such as question answering, event extraction, and entailment detection [117, 141].

Previous work that tackled the problem of identifying synonymy between relational phrases employed similarity metrics based on textual strings [159] or dependency paths [71, 83, 94] of the two relational phrases. Kok and Domingos [66] proposed a probabilistic model based on two Markov logic networks (MLNs) [113] to simultaneously cluster objects and relations. Nebot and Berlanga [97] used a probabilistic model inspired by statistical machine translation to cluster relations in biomedical documents. These models are unsupervised in the sense that no manual labeling of clusters by human is needed. One of the major shortcomings of their approaches, however, is that they only focus on the textual surface of arguments of a relation to estimate the synonymy probability and cannot effectively capture other features, such as the context around the relations.

To address the above shortcoming, we apply the continuous bag-of-words (CBOW) model, a deep-learning technique proposed by Mikolov et al. [80], to represent our relational phrases. A relation in the format of <entity 1, relational phrase, entity 2> is identified in a sentence, and each of the two entities and relational phrase is regarded as a newly defined *word*. We thus treat the entities and the phrase differently from the other words depending on their corresponding roles in the relation. The CBOW model then learns the distributional representations of the relational phrases through a feed-forward neural network language model [9], which allows us to capture the context around a relational phrase when learning its representation.

Sun and Korhonen [135] also used the context around verbs for the task of verb classification by introducing a rich set of semantic features. The features include collocations of verbs, prepositional preference, and lexical preference in subject, object and indirect object relations. The key difference between their work and ours is that we cluster verbs and verb phrases that compose biomedical relations while they only focus on single verbs.

We have compared our approach with three unsupervised methods: bag-of-words (BOW), latent Dirichlet allocation (LDA) [15], and Semantic Network Extractor (SNE) [66]. Regarding BOW and LDA, we treat a relational phrase as a *document* (in LDA terms) and entities that share the same phrase as *words* in the document. The BOW model represents each relational phrase as a sparse vector of occurrence counts of entities. LDA-SP [122], which is developed from LinkLDA [34] to model selectional preferences, simultaneously models two sets of distributions for two entities of a relation. Each entity is drawn from a hidden topic. LDA-SP assigns a higher probability to the state in which the two hidden topics are equal. For each relational phrase, the model outputs a vector of the prior topic distribution. We then apply the k-means algorithm on top of vector representations to cluster phrases into synonymous groups.

SNE tackles the task of clustering relational phrases by a probabilistic model trained on two MLNs. Unlike the other methods, SNE performs clustering on a database of relations, i.e., it does not consider the context or the frequency of relations. However, SNE can automatically identify the best number of clusters and simultaneously cluster objects and relational phrases.

We have conducted experiments using a large set of biomedical relations extracted from MEDLINE by our pattern-based open information extraction system presented above. The results show that word embeddings significantly outperform BOW, LDA-SP and SNE. They can boost the performance of clustering by more than 10% of F-score compared with the other methods. In addition, we demonstrate how the obtained clusters of relational phrases could be used to improve the performance of high-level text-mining applications such as question answering and entailment detection.

1.3 Dissertation Outline

The remainder of this dissertation is organized as follows.

- **Background (Chapter 2)**. This chapter first describes details about text mining. Next, the background of biomedical text mining, including the two important tasks of named entity recognition and relation extraction, and the available corpora available in this field, are presented. Then, approaches of unsupervised biomedical relation extraction and some notable systems are mentioned in this chapter. We also present an overview of predicate-argument structures. Some related work and methods of synonym resolution for relational phrases are described in detail.

- **Binary Relation Extraction for Biomedical Texts (Chapter 3)**. Details about our pattern-based system, such as noun phrase pairs detection and named entity recognition, are shown in this chapter. We then present the principles of manual evaluating extracted relations. We also describe two types of our evaluation scenarios, which is evaluating general and pre-defined relations. Comparison of our system and other state-of-the-art ones are reported. The output of our system on the whole MEDLINE corpus are also discussed in this chapter.

- **Synonym Resolution for Relational Phrases (Chapter 4)**. This chapter describes three methods used for modeling the relations: bag-of-words, topic models, and word embeddings. The original LDA-SP and our modified model, LDA-SP-sem, are presented in detail. We also describe three ways of representing a relation when using word embeddings. Finally, we report the experimental results and discuss the usability of the obtained clusters in high-level text-mining applications.

- **Conclusion and Future Work (Chapter 5)**. We give conclusions on our research by clarifying its advantages and disadvantages. Based on the conclusions, we will propose some future work.

Chapter 2

Background

2.1 Text Mining

Text mining can be defined as a process that aims at extracting interesting and non-trivial patterns or knowledge from unstructured textual data in document collections. There are two points that distinguish text mining from data mining. First, as mentioned above, the data source of text mining are unstructured textual data, while data mining assumes that the data have already been in a structured format. Second, text mining systems involve natural language processes to convert unstructured documents into more explicitly structured data, which is not a concern in data mining systems.

Generally, a text mining system performs six main tasks as depicted in Figure 2.1. These tasks are (1) sentence segmentation, (2) tokenization, (3) part-of-speech tagging, (4) parsing, (5) named entity recognition (NER), and (6) relation extraction. The first four tasks are general purpose natural language processing (NLP) tasks, while NER and relation extraction, tasks of information extraction, are problem-dependent. In this section, we will present an overview of the two tasks in the general domain.

2.1.1 Named Entity Recognition

Named entity recognition (NER) is to identify and classify all phrases which refer to entities, mostly things in the real world, of specified semantic types. In general NER systems, some common entities such as people, places, organizations, date and time [36] are detected as illustrated in Figure 2.2. While specialized applications may be concerned with other types of entities, e.g., percentage, monetary, products, works of art, genes, proteins and other biological entities [6].

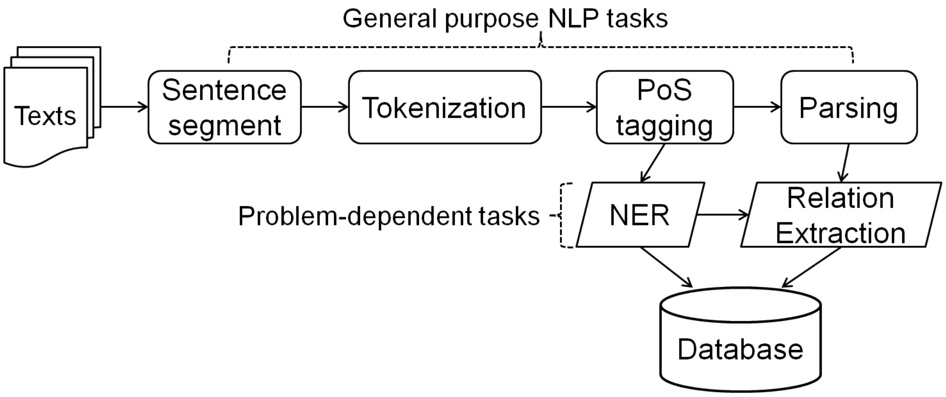


Figure 2.1: An overview of text processing tasks.

[$_{ORGANIZATION}$ **U.N**] official [$_{PERSON}$ **Ekeus**] heads for [$_{LOCATION}$ **New York**].

Figure 2.2: Examples of NER in general domain.

A standard way to perform NER is assigning word-by-word tags that capture both the boundary and the type of the named entities [56, 142]. For instance, named entities in Figure 2.2 can be represented by the B-I-O representation shown in Table 2.1. In this scheme, a *B* label indicates the beginning word of an entity, an *I* represents the word inside the entity, and an *O* indicates a word outside any entity. By using this scheme, NER is converted into sequence labeling of words in a sentence. Next, we need to design a set of features to label those words.

Nadeau and Sekine [95] classified common NER features into three categories: word-level features, list lookup features and document and corpus features. Word-level features are defined over the lexical terms composing an entities, e.g., the textual surface, part-of-speech, the chunking tag, and the shape feature of a token as shown in Table 2.1. List lookup features are extracted from publicly available resources, such as a list of place names – gazetteers, dictionary of protein names, stop words, and typical words in organization names. Instead of using available lists or dictionaries, we can perform clustering on a large data to group words that have similar senses or meanings into clusters and use these clusters as lists to extract features for NER. Document features are extracted from the content and the structure of the document containing targeted entities. For instance, we can count the occurrences of a word in uppercased and lowercased forms in a single document. We can also extract meta-information from the document, such as the fact that news often start with a location name or email headers are good indicators of person names.

Given a training set with the above-mentioned extracted features, we can train a classifier

Table 2.1: Representing named entities by the B-I-O style with some word-level features.

Words	Features				Label
	Text	POS	Chunking	Shape	
U.N	u.n	NNP	B-NP	upper	B-ORG
official	official	NN	I-NP	lower	O
Ekeus	ekeus	NNS	I-NP	cap	B-PER
heads	heads	VBZ	B-VP	lower	O
for	for	IN	B-PP	lower	O
New	new	NNP	B-NP	cap	B-LOC
York	york	NNP	I-NP	cap	I-LOC
.	.	.	O	punc	O

to label sequential words by using any supervised learning methods, e.g., Support Vector Machines (SVMs) [125], Conditional Random Fields (CRFs) [74] and Hidden Markov Model (HMM) [10].

To evaluate a NER system, we need a gold standard, a text data that is annotated by human, and a metric that relates the gold standard and the system's output. So far, there are several metrics used at MUC, CoNLL and ACE conferences. In the named entity task of MUC-7, a system is scored based on the correct type (TYPE) and the exact text (TEXT) generated by the system [21]. An entity is counted as a correct TYPE if it is assigned the correct semantic class regardless of its boundaries as long as there is an overlap with the gold standard entity. An entity is counted as a correct TEXT if its boundaries are correct regardless of the type. The precision is computed by the fraction of correct answers and the number of entities that a system detects. The recall is computed by the fraction of correct answers and the number of all entities in the gold standard. The final score is the harmonic mean (F-score) of precision and recall. The highest performance of MUC-7 was an F-score of 93.39% [79].

CoNLL used an exact-match evaluation, which is stricter than MUC-7. An entity is counted as correct if its type and boundaries are exactly matching with the gold standard entity. CoNLL also employed the same definition for the precision and recall as MUC-7. The highest performance at CoNLL 2003 was about 92% F-score for PERSON and LOCATION entities, and about 84% for ORGANIZATION [142].

The ACE score [33] is defined by the sum of the value for each system output entity

Affiliation

$[_{ORGANIZATION}$ **U.N**] official $[_{PERSON}$ **Ekeus**] heads for $[_{LOCATION}$ **New York**].

Figure 2.3: Examples of relation extraction.

accumulated over all system output. The system output entity is calculated as:

$$Value_{sys\_entity} = Entity\_Value(sys\_entity) \times \sum_{m} Mention\_Value(sys\_mention_m), \tag{2.1}$$

where $Entity\_Value$ and $Mention\_Value$ are functions[1] that score entities or mentions based on their matching to the gold standard.

Among the three metrics, the MUC and CoNLL are intuitive and easy to implement, while the ACE is complex and may make the error analysis difficult.

2.1.2 Relation Extraction

The goal of relation extraction is to detect occurrences of a pre-specified type of relationship between a pair of entities of given types, e.g., affiliation (persons to organizations), interactions between genes/proteins, physiological process between proteins and cells. An example of relation extraction based on the named entities from Figure 2.2 is shown in Figure 2.3. The relation tells us that 'Ekeus' is an officer of the 'U.N' organization.

Approaches to relation extraction can be divided into two categories: supervised and semi-supervised approaches.

The supervised methods require a gold standard, i.e., a small data that is annotated by human analysts, to learn their model. A simple way is to divide relation extraction into two subtasks: (1) detect whether there is a relationship between two entities, and (2) classify the detected relation into pre-defined categories, e.g., affiliation, directional, and part-of relations. The first subtask can be addressed by training a classifier to decide whether a given pair of entities is true or false. The true pairs are extracted based on the gold standard, while the false pairs can be generated from entities in the same sentence that do not compose relations according to the annotated data. The second subtask can also be implemented by training a classifier with multiple classes. Each class corresponds to a category of relations.

Similarly to NER, after modeling the problem, the next step is to select suitable features

[1] More detail about these functions is available at http://www.itl.nist.gov/iad/mig/tests/ace/2004/doc/ace04-evalplan-v7.pdf

Table 2.2: Samples of features for the relation <U.N, Ekeus>.

Features of named entities	
Entity 1 type	ORG
Entity 1 head	U.N
Entity 2 type	PER
Entity 2 head	Ekeus
Features from words	
Word-distance between the two entities	1
Number of entities between the two entities	0
Syntactic features	
Dependency-tree paths	U.N $\leftarrow_{nn}$ Ekeus
Chunk base-phrase paths	NP

for the classifiers. Jurafsky and Martin [56] categorized the common features into three classes as follows:

- Features of named entities, such as headwords or types of the two entities
- Features from the words in the text, such as stemmed words, distance in words between the two entities, and number of entities between the two entities
- Features from the syntactic structure, such as chunk base-phrase paths, dependency-tree paths, and constituent-tree paths.

Samples of features for the relation <U.N, Ekeus> in Figure 2.3 are presented in Table 2.2. After extracting features, we can train the classifiers by using conventional machine learning methods, e.g., SVM [18, 46].

By using supervised methods, the system can achieve high precision and recall. However, the annotated corpora are not available for all types of relations, and fully supervised methods are not applicable to large-scale relations of Web texts. Another promising approach, called semi-supervised, can create additional features or training data based on prior knowledge, e.g., ground facts, and a large unlabeled data.

Sun et al. [134] employed word clusters as additional features for relation extraction. They then proposed several statistical methods to select effective clusters, such as calculating the clusters' information gain or coverage to decide whether a cluster is appropriate feature or not. They used the English portion of the TDT5 corpora (LDC2006T18) as unlabeled data for inducing word clusters. The experimental results on the benchmark ACE 2004 training data showed that the approach outperformed the supervised relation extraction system by Zhou et al. [46].

Another semi-supervised technique that has recently attracted many researchers in relation extraction is distant or weak supervision [53, 52, 84, 116]. This approach generates its own training data by matching a set of facts from available knowledge-bases (e.g., YAGO [133] and Freebase [17]) to large unlabeled texts. The distant supervision assumes that if two entities participate in a relation, any sentences that contain the two entities might express that relation [84]. For instance, suppose that $r(e_1, e_2) = Founded(Jobs, Apple)$ is a ground tuple in a knowledge-base and $s =$ "Steve Jobs founded Apple, Inc." is a sentence containing synonyms for both $e_1 =$ Jobs and $e_2 =$ Apple, then s may be a natural language expression of the fact that $r(e_1, e_2)$ holds and could be a useful training example [52]. The generated training data is then used to learn a relation extractor.

There are two separate methods to evaluating systems of relation extraction [56]. The first method calculates both labeled and unlabeled recall, precision and F-score by exactly matching the generated results with the gold standard. The labeled scores measure the system's ability to classify relations while the unlabeled ones focus on detected entities. The second method computes the scores based on the extracted tuples rather than on the relation mentions, i.e., it ignores the frequencies of a relation in the text.

2.2 Biomedical Text Mining

Biomedical text mining, also known as biomedical natural language processing (BioNLP), is a narrow field of text mining that aims to develop methods for extracting useful information from the literature in biomedical and biology domains. Therefore, biomedical text mining also follows the same process as text mining. The only difference is that the process is applied to biomedical documents, which requires us to make it suitable for the domain. This section will describe more details about the difference in the two main tasks: biomedical NER and relation extraction.

2.2.1 NER

Compared with general NER, there are some difficulties for biomedical NER as follows:

- Long entity: An entity may be very long, e.g., 'isolated peripheral blood mononuclear cells' and 'oxidative-stress sensitive transcription factor'.

- Irregular name conventions: An entity may be written in various forms, such as 'Geobacillus sp. strain T1', 'Geobacillus sp. T1', and 'Geobacillus Strain T1'. There is no convention for presenting the short form of an entity; sometimes they are full short form, but sometimes they are not. For instance, 'B. cereus' is a short form of the bacteria 'Bacillius cereus'; 'M. TB.' is a short form of 'Mycobacterium tuberculosis'.

- Nested named entity: An entity contains another entity, e.g., the RNA entity 'CIITA mRNA' includes the DNA entity 'CIITA'.

Due to its importance to all further processing, NER is one of the most widely studied tasks in biomedical text processing. The early NER systems in the field are typically rule-based or lexicon-based [6, 39, 40, 41, 96, 121, 138]. MedLEE is a general natural language processor for clinical texts, encoding and mapping terms to a controlled vocabulary [39]. GENIES [40], adapted from MedLEE, identifies genes and proteins by using BLAST techniques, specialized rules and external resources, including GeneBank and Swiss-Prot. EDGAR [121] extracts information about drugs and genes relevant to cancer from the biomedical literature based on their semantic and pragmatic analyses. The advantage of these systems is that they do not need labeled data to be trained and they are applicable to large-scale texts.

After its releasing, the GENIA corpus [61] has been used for various supervised learning models, including SVM [57, 85, 164], HMMs [163], and CRFs [76, 129, 161]. Those studies converted NER to sequence labeling and employed the common NER features as mentioned above to address non-nested entities. Nested entities were tackled by modeling the recognition as a parsing problem [37] or by reducing the nested problem to one or more BIO problems to make use of existing NER tools [3].

It should be noted that despite the availability of training corpora, the performance of biomedical NER has not been as high as expected. The shared task of BioNLP/NLPBA 2004 used GENIA as dataset for training and evaluation [62], and the highest-performing system only achieved 72.6% of F-score. In the first BioCreative challenge [51], gene mention identification was the first subtask of task 1 and the highest F-score was about 83% [160]. In the BioCreative II, the F-score was improved to 87% [5], and even a combined system assembled by the organizers achieved an F-score of only 90.66% [54].

Most of the previous work has focused on some specific semantic types, such as the gene/protein names [64, 92, 129, 145, 160], gene/protein, cell line and cell type [57, 62, 161, 164], drugs [121, 147], and chemicals [96, 147]. There are only a few studies that tackle a general schema of biomedical entities, e.g., MetaMap[6] and CubNER [162]. MetaMap maps biomedical text to UMLS Metathesaurus concepts. The system first performs lexical and syntactic analysis (general NLP processes) on the input text, and then applies some matching techniques to match the input words with Metathesaurus strings. Pratt and Yetisgen-Yildiz [104] reported that MetaMap achieved a recall of 52.8% and a precision of 27.7% with exact matching, and a recall of 93.3% and a precision of 55.2% with partial matching when applying on 20 MEDLINE titles. CubNER [162] is an unsupervised biomedical NER that has three main steps: (1) collecting seed terms from the UMLS Metathesaurus, (2) detecting candidates of entities based on their own heuristic, and (3) calculating the similarity between the candidates and the seed terms to classify the entities. The system achieved 53.1%, 52.2%, 53.9% and 39.5% F-score on the Pittsburgh, Beth, Partners and GENIA corpus, respectively.

2.2.2 Relation Extraction

Two series of shared tasks including BioCreative [51] and BioNLP shared tasks [60] have significantly contributed to the progress of biomedical relation extraction. The BioCreative tasks focused on protein-protein interaction (PPI), while the BioNLP shared tasks have aimed at bio-molecular events.

A variety of different approaches have been proposed to solve PPI and bio-molecular events. Airola et al. [1] constructed a kernel for learning from dependency graphs of sentences to extract PPI. Miwa et al. [87] proposed a method that combined a bag-of-words kernel, subset tree kernel and graph kernel to capture important information from given sentences and applied it to PPI extraction. They achieved better performance than the Airola et al.'s system. Björne et al. [12] represented the relationship between entities as a graph; each node can be an entity or even an event. The semantic graph would be updated with more nodes and edges by trigger and event detection. The graph was then post-processed by some heuristics to generate the final semantic one, which represented for event extractions. Their system achieved the highest performance at the BioNLP shared task 2009. The author, then, extended their system [13] to generalized biomedical domains by using an SVM classifier at the post-processing step instead of heuristics.

Riedel et al. [114] applied Markov Logic Networks (MLNs) to extract the event and achieved the best results for negation and speculation events. The highest-performing system in the 2013 shared task was TEES-2.1 [14], which achieved an F-score of about 55% for the CG task. The system represents both binary relations and events with a unified graph and approaches event extraction task as a mutli-class classification task (SVM^{multiclass}). Liu et al. [152] and Roller et al. [123] also employed mutli-class classification for their systems. Other methods such as domain adaptation [88, 115] and dual composition [115] were also applied to this task.

The above approaches require gold standard corpora (i.e. manually annotated corpora) for their training step. It is therefore difficult to apply the learned models to other tasks or to scale them to environments of large documents. Pattern-based methods can tackle this limitation [42, 109, 118, 153]. Rinaldi et al. [118] introduced three levels of patterns to detect relations. The first level is syntactic patterns; the second one is semantic rule, which normalizes many possible syntactic variants (e.g. active, passive, nominalizations). On the third level they combined semantic rules with lexical and ontological constraints to obtain specialized queries that can detect a domain-specific relation.

RelEx [42] extracted protein-gene interactions by using three crafted rules. Their rules were constructed based on the dependency parse tree of a sentence. The system then filtered out candidates of relations by performing some post-processes including negation check, effector-effectee detection, enumeration resolution and restriction to a set of relation terms. RelEx was applied to about 1 million MEDLINE abstracts, and extracted about 150,000 relations with an estimated performance of both 80% precision and 80% recall. Recently, Xu et al. [153] proposed a pattern-learning approach to extract treatment-specific drug-disease. They first identified drug-disease pairs according to drug and disease lexicons from the UMLS Metathesaurus and DrugBank. They then learned patterns of treatment based on some initial extracted ones. Their system finally detected 34,305 unique drug-disease treatment pairs from the whole MEDLINE. Though these pattern-based approaches can be implemented without a labeled corpus, they are designed to a specific type of relations, not to a wide range of types. In order to address this limitation, we will propose a set of patterns that covers a wide range of relation types. More related work to this approach will be presented in the following sections.

2.2.3 Available Resources and Corpora

A variety of biomedical resources and corpora have been published and made available to the research community. In this section, we will introduce some notable ones that are often used in BioNLP.

MEDLINE/PubMed

MEDLINE/PubMed is a primary source for input documents in BioNLP. It is a collection of documents but it is not static since it grows every year. MEDLINE is a bibliographic database containing citations and author abstracts from more than 5,600 biomedical journals. Currently, MEDLINE contains over 21 million references generally from 1946 to the present[2]. One MEDLINE citation represents one journal paper and includes some fields, such as title, authors, and abstract. PubMed[3] is a search engine providing access to the MEDLINE database. PubMed indexes each MEDLINE citation and assigns a PubMed Unique Identifier (PMID) to it.

So far, there are many text mining systems applied to the whole MEDLINE, such as MEDIE [89], Björne et al. [11], MedScan [29], AliBaba [102], and SemMedDB [59].

The Unified Medical Language System

The Unified Medical Language System[4] (UMLS) is a large set of lexical resources that integrates key terminology in the clinical domain. The UMLS provides three knowledge sources: the UMLS Metathesaurus, the Semantic Network and SPECIALIST Lexicon and Lexical Tools. The UMLS Metathesaurus is a dictionary of terms from many vocabularies. Currently, the 2014AA version contains more than 2.9 million concepts and 11.6 million unique concept names from over 150 source vocabularies. The Semantic Network is a hierarchy of 54 semantic relations between 133 semantic types. The semantic types in the Semantic Network are consistent with categories of all concepts in the UMLS Metathesaurus. SPECIALIST Lexicon and Lexical Tools is a set of NLP tools that support lexical variation and text analysis tasks in the biomedical domain.

The Open Biomedical Ontologies

The Gene Ontology (GO)[5] is one of the first open ontologies in the field. The GO provides a controlled vocabulary of terms representing gene product properties. It covers three domains: (1) cellular component describes locations of sub-cellular structures and macromolecular complexes; (2) molecular function describes activities such as binding activities at the molecular level; and (3) biological process, a recognized series of events or molecular

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

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

[4] http://www.nlm.nih.gov/research/umls/

[5] http://www.geneontology.org/

functions.

Other ontologies for phenotype, biochemistry and anatomy structured similarly to GO can be found at the Open Biomedical Ontologies website: http://www.obofoundry.org/.

The BioLexicon

The BioLexicon[6] is a lexical resource that combines terminologies of several public data resources such as UniProtKb[7], ChEBI[8] and NCBI taxonomy[9]. The resource [141] contains over 2.2 million of both domain-specific and general lexicon entries with information of four part-of-speech (POS) categories (nouns, verbs, adjectives and adverbs), and over 1.8 million terminological variants, as well as over 3.3 million semantic relations with over 2 million synonymy relations.

Applications of BioLexicon to lemmatisation of biomedical text, information retrieval, and information extraction were reported in Thompson et al. [141].

DrugBank

The DrugBank database [69] is a unique bioinformatics and cheminformatics resource that combines detailed drug (i.e. chemical, pharmacological and pharmaceutical) data with comprehensive drug target (i.e. sequence, structure, and pathway) information. The database contains 7,681 drug entries including 1,545 FDA-approved small molecule drugs, 155 FDA-approved biotech (protein/peptide) drugs, 89 nutraceuticals and over 6,000 experimental drugs. Additionally, 4,218 non-redundant protein (i.e. drug target, enzyme, transporter, or carrier) sequences are linked to these drug entries. Each DrugCard entry contains more than 200 data fields with half of the information being devoted to drug/chemical data and the other half devoted to drug target or protein data.

GENIA

The GENIA corpus [61] is an annotated corpus for molecular biology. The corpus consists of 2,000 MEDLINE abstracts with more than 400,000 words. It is fully annotated with both linguistic and semantic markups, including sentence boundaries, token boundaries, POS tags, and named entities of 47 biologically categories. Part of the corpus were annotated

[6] http://www.ebi.ac.uk/Rebholz-srv/BioLexicon/biolexicon.html

[7] http://www.uniprot.org/

[8] http://www.ebi.ac.uk/chebi/

[9] http://www.ncbi.nlm.nih.gov/taxonomy

with syntactic trees, coreference resolution and biomolecular events, which have been used in the BioNLP shared task.

The GENIA corpus has been demonstrated to be the most heavily used corpus in the BioNLP community, as described in Section 2.1.1 and Section 2.1.2.

Five Protein-Protein Interaction Corpora

In biomedical relation extraction, protein-protein interaction (PPI) has been the most widely studied relation because of its important role in biological processes. There are five commonly used corpora of this relation type, including AIMed, BioInfer, HPRD50, IEPA and LLL. AIMed[10] was created from 200 PubMed abstracts containing PPI and 30 abstracts with no PPI. BioInfer[11] consists of about 1,100 sentences from PubMed abstracts that contain at least one pair of interacting proteins. HPRD50[12] consists of 50 abstracts referenced by the Human Protein Reference Database (HPRD). The IEPA corpus [32] was constructed from about 300 PubMed abstracts, each abstract contains at least two biochemical nouns. The LLL[13] corpus, consisting of only 55 sentences, was the shared dataset for the Learning Language in Logic 2005 challenge. The domain of LLL is gene interactions of Bacillus subtilis.

All the five corpora contain annotation of identifying genes or proteins. Among them, only LLL and BioInfer contain information on the types of the entities, such as ‘individual protein’ and ‘protein complex’. Moreover, these two corpora also distinguish the types of PPI, e.g., ‘positive action binding’ and ‘positive action cross-link’ in BioInfer, and ‘explicit action’ and ‘Binding to Promoter’ in LLL, while the others only label interactions. Pyysalo et al. [105] provided a software tool necessary to convert those corpora into a shared XML-based format.

2.3 Open Information Extraction

Open Information Extraction (Open IE) has become prevalent over traditional relation extraction methods, especially on the Web. The idea of Open IE is to avoid the need for specific training examples and to extract a diverse types of relations. More details about Open IE systems for Web text and for biomedical domain will be presented in this section.

[10] ftp://ftp.cs.utexas.edu/pub/mooney/bio-data/

[11] http://mars.cs.utu.fi/BioInfer/

[12] http://www2.bio.ifi.lmu.de/publications/RelEx/

[13] http://genome.jouy.inra.fr/texte/LLLchallenge/

2.3.1 General Domain

Banko et al. [8] introduced Open IE as a novel information extraction paradigm that facilitates domain independent discovery of relations extracted from text and readily scales to the diversity and size of the Web corpus. An Open IE system extracts tuples consisting of argument phrases (arg1, arg2) from the input sentence and a relational phrase (rel) that expresses the relation between arguments, in the format of (arg1; rel; arg2). Open IE systems that have been developed up to now include TextRunner [8], StatSnowBall [165], WOE [151], ReVerb [35], and OLLIE [73].

TextRunner [8] consists of three modules, including Learner, Extractor and Assessor. The Learner first applies a parser to sentences of its own training data to detect candidate tuples $(e_i, r_{i,j}, e_j)$, in which e_i, e_j are base noun phrases. It then assigns each tuple as true or false based on some syntactic constraints. Finally, a Naive Bayes classifier was learned on these extracted tuples. The Extractor extracts candidate tuples from input sentences by using some heuristics and sends the tuples to the classifier, if the tuple is validated as true, it would be passed to the Assessor. Finally, the Assessor assigns a probability to the tuple. TextRunner was applied to a corpus consisting of over 9,000,000 Web pages and has shown the ability of extracting a broader set of facts.

The WOE systems [151] also approached Open IE in the same way as TextRunner. However, they made use of Wikipedia as a source of training data for their extractors, which led to further improvement over TextRunner. In addition to traditional relation extraction, StatSnowBall [165] also addressed Open IE on Web text. They used the discriminative MLNs [103] to learn the weights of their generated patterns and applies some softened hand rules to assign the weights.

Fader et al. [35] proposed ReVerb to overcome two shortcomings in Open IE systems: incoherent extractions and uninformative extractions. ReVerb introduced a syntactic constraint to validate incoherent extracted relations, and a lexical constraint to avoid overly-specific relation phrases. Their system achieved an area under the curve that is 30% higher than WOE or TextRunner.

Since ReVerb focuses on relations mediated by verbs (verb, verb + preposition, verb + noun + preposition), OLLIE [73] is proposed to extract other relations mediated via nouns and adjectives. First, it uses a set of high precision seed tuples from ReVerb to bootstrap a large training set. Second, it learns open pattern temples over this training set. Next, OLLIE applies these pattern templates at extraction time. Both ReVerb and OLLIE assign a confidence value to each extracted triple, instead of simply classifying them as true or false.

TreeKernel, a more general method than the above systems was presented by Xu et al. [154]. They employ multiple SVM models with dependency tree kernels for their two tasks: determining if a sentence potentially contains a relation between two entities and confirming explicit relation words for those entities. The shortest path between the two entities along with the shortest path between relational words and an entity are considered as a candidate tree path and input to a tree kernel. They finally used kernel-based SVMs to classify a relation triple as true or false.

Recently, Mesquita et al. [78] proposed EXEMPLAR to identify both binary and n-ary relations. EXEMPLAR employed six patterns based on dependency trees to extract n-ary relations. Their experimental results implicated substantial gains over both binary and n-ary relation extraction tasks compared with REVERB, OLLIE and TreeKernel.

2.3.2 Biomedical Domain

SemRep [119, 120], a rule-based semantic interpreter, extracts semantic relationships from biomedical text. Their relationships are represented as *predications*, a representation consisting of a predicate and two arguments. SemRep extracts 30 predicate types, mostly related to clinical medicine, substance interactions, genetic etiology of disease and pharmacogenomics. SemRep relies on 'indicator' rules which map verbs and nominalizations to predicates in the Semantic Network, such as TREATS, AFFECTS and LOCATION_OF. For example, an indicator rule says that the nominalization *treatment* must be mapped to the predicate TREATS. SemRep also enforces domain restrictions by using meta-rules that require all semantic relations to be present in the Semantic Network. For instance, a pair of semantic types that matches to the predicate TREATS is 'Pharmacologic Substance' and 'Disease Syndrome'. Therefore, the arguments associated with *treatment* for example, must have been mapped to the Metathesaurus concepts with the semantic types of 'Pharmacologic Substance' and 'Disease Syndrome'. Consequently, for each type of relations, SemRep has to refine the corresponding 'indicator' rules and meta-rules based on the the UMLS Semantic Network. Regarding this point, our patterns are more general than SemRep, since they are tailored to capture deep syntactic relations and not restricted to any specific set of verbs.

Rosemblat et al. [124] have recently extended SemRep's coverage to the field of medical informatics. They adapted ontology engineering processes to build a semantic representation of an unsupported domain, and then integrated it with the UMLS Metathesaurus so that SemRep can be applied to the new domain. They conducted some experiments to compare

Table 2.3: Lexico-syntactic patterns by Nebot and Berlanga [97].

Pattern	Examples
[E] verb [E]	[levamisole] activates [macrophrages]
[E] verb phrase [E]	[PAF] consistently inhibited [killer cell]
[E] verb phrase + prep [E]	[polysaccharide] was treated with [periodate]
[E] prep + noun + prep [E]	[cytostatic drugs] in combination with [OK-432]
[E] to + infinitive [E]	[fibroblasts] to produce [growth factor(s)]
[E] neg-verb-phrase [E]	[haptens] does not inactivate [B lymphocytes]
[E] to be [E]	[Strongyloidiasis] is an [intestinal disease]

SemRep and the enhanced SemRep, their results have shown that the enhanced version performed better than SemRep in terms of precision.

McIntosh et al. [77] presented a bootstrapping system that does not use manually-crafted seeds of tuple or pattern. The system first identifies the terms in the target categories by using some hand-picked seed terms. Next, their relation discovery module automatically finds the relation and their seeds based on some heuristics and sends the terms back to the term recognition module. Their system was applied to MEDLINE abstracts to extract relations between 10 categories of entities and achieved high precision, the highest one was 87.9%.

The system by Nebot and Berlanga [97] extracts explicit binary relations of the form <*subject, predicate, object*> from CALBC [110] initiative. To detect candidate relations, they proposed seven simple lexico-syntactic patterns as shown in Table 2.3. These patterns are expressed in part-of-speech tags in which relational phrases reside between the two entities. By contrast, our PAS patterns do not restrict the order of relational phrases and arguments in sentences. This means that our system can detect more relations than Nebot and Berlanga's system.

2.4 Predicate-Argument Structures

A relation is described in a sentence by a composition of a predicate and its arguments, which forms Predicate-Argument Structure (PAS). A predicate that indicates a particular type of relation can be a verb, a phrase or a preposition. Figure 2.4 shows an example of a PAS in a sentence. In this example, 'activates' is a predicate and 'LPS' and 'macrophages'

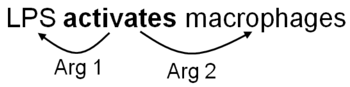


Figure 2.4: An example of predicate-argument structures.

are its two arguments, in which ‘LPS’ is the subject and ‘macrophages’ is the object. The verb ‘activate’ indicates the relationship of interaction between the two arguments. In this work, six PAS patterns that we propose focus on verb and preposition predicates.

Using PASs has been a practical approach to domain-independent information extraction. Gildea and Palmer [43] and Surdeanu et al. [136] built systems that automatically identified PAS on the output of a full parser by using machine learning methods. Some annotated corpora of PAS frames such as PropBank [63], VerbNet [65], and FrameNet [7] were employed in their learning models. These detected PASs were then applied to extract information in general domains.

In the biomedical domain, Tateisi et al. [139] introduced a corpus annotated with PAS in verbs including their normalized forms on research abstracts. PASBio [150] and BioProp [22] are PAS frames for the biomedical domain based on PropBank. PASBio was applied to the LSAT system [130] to extract alternative transcripts from the same gene. BioProp was used to train the BIOSMILE system [143], a system that performs semantic role labeling on biomedical verbs. However, these two resources are not general enough for the domain since both of them are restricted to a limited number of verbs. More specifically, BioProp contains 2,382 predicates for 30 biomedical verbs; and PASBio has only 30 predicates[14] for 30 verbs describing molecular events[15]. In this dissertation, we propose six simple PAS patterns that are expected to cover both BioProp’s and PASBio’s two-argument frames.

Instead of using hand-written patterns, Yakushiji et al. [156] automatically constructed their PAS patterns based on the obtained syntactic structures. They then enhanced their system [155] by dividing the patterns into combination patterns and fragmental patterns, and learning a SVM prediction model based on the pattern matching results and scores. The best performance of their system was an F-score of 57.3%. However, their system was applied to PPI only since it was trained on the AIMed corpus.

Subcategorization frames (SCFs) [117] are similar to PASs. However, compared with PASBio and BioProp, SCFs are more general in the sense that their arguments are not restricted to a specific type but built from phrase types. For instance, a PAS for the verb

[14] https://sites.google.com/site/nhcollier/projects/pasbio/all-predicates

[15] The 30 biomedical verbs of BioProp are different from the 30 verbs of PASBio.

‘mutate’ in PASBio is:

Arg1: physical location where mutation happen //exon, itron//
Arg2: mutated entity //gene//
Arg3: change at molecular level
Arg4: change at phenotype level

While SCFs for the verb ‘decrease’ are NP (the verb has one argument that is a noun phrase, NP-PP (the verb has two arguments: a noun phrase and a prepositional phrase) and PP-PP (two arguments are both prepositional phrases). Those frames are less restrictive than the PAS frames.

In this dissertation, our proposed PAS patterns are more general than SCFs since our patterns are not restricted to any specific verbs and accept all verbal forms.

2.5 Synonym Resolution for Relational Phrases

The task of finding synonyms for extracted relations is usually known as synonym resolution, or paraphrase discovery. Several unsupervised systems for this task have focused on using similarity-based, corpus-based and probabilistic techniques.

2.5.1 Similarity-based Methods

The DIRT system [71] uses a similarity measure based on mutual information statistics to identify inference rules between relation phrases. The authors assumed that if two dependency paths tend to link the same sets of words, their meanings are similar. In their context, a path represents a binary relationship, from each pair of similar path, an inference rule is generated. To compute the path similarity using their assumption, they collected frequency counts of all paths and slot fillers for the paths in a corpus. Each slot filler is considered as the path’s features. For instance, they calculated the similarity between “X finds a solution to Y” and “X solves Y” based on the instances of the slots X and Y, with an assumption that if the two paths tend to occur in similar contexts, i.e., the same instances of X and Y, the meanings of the paths tend to be similar.

Min et al. [83] employed the same approach as DIRT to discover paraphrases in WEBRE. However, there are two differences in their work. First, they measured the similarity between two relational phrases instead of between two dependency paths. Second, features of the relational phrases were ordered pairs of instances of the two slots X and Y while DIRT did not consider the order of the instances. Their empirical tests showed that WEBRE could produce likely paraphrases but DIRT could detect general inference rules.

Hasegawa et al. [48] also employed a similarity metric in their method. The method first calculates the cosine similarity among pairs of named entities based on the context words present between the two entities. They then used a hierarchical clustering technique to cluster those pairs. RESOLVER's [159] uses the same method but the similarity is computed based on a formal probabilistic model.

2.5.2 Corpus-based Methods

Sekine [128] proposed an unsupervised method to discover paraphrases from a large unlabeled corpus, without using any seed phrase. They first extracted Named Entity (NE) pairs from the corpus and found keywords for each NE pair. The top-ranked word based on the TF/IDF metric was selected as the keyword of a pair. Finally, all phrases that have the same keyword are clustered in the same group. If the same pair of NE is used in different groups, those groups will be linked together. They conducted experiments on newswire corpora; depending on the evaluated domain, the accuracy of paraphrasing was from 73% to 99%.

Davidov and Rappoport [31] presented another unsupervised method to find groups of relation patterns. As a first step, a set of hook words, i.e., words whose frequency is not higher than F_C and not lower than F_B, are selected. For each hook word, a hook corpus is created by extracting a set of contexts that contain the hook word. Those corpora are then filtered based on their general patterns and some heuristics. The resulting corpora are considered as pattern clusters. They then merge those clusters by their own clustering algorithm. Their evaluation has shown a similarity between their clusters and human notions.

2.5.3 Probabilistic Methods

Nebot and Berlanga [97] employed a statistical model for their clustering algorithm. The algorithm takes the candidate relations $<s_1, r_s, s_2>$ and calculates the synonymy probability of two relation strings (r_1, r_2) based on the semantic types (T_d, T_r) of the head entities of s_1 and s_2. The joint probability, inspired by a statistical translation method, can be estimated as follows:

$$p(r_1 \sim r_2 | (T_d, T_r)) \propto \sum_{(e_1,e_2)\in(T_d,T_r)} p(r_1|(e_1, e_2)) \cdot p(r_2|(e_1, e_2)) \cdot p((e_1, e_2)) \qquad (2.2)$$

Finally, to check whether two relation strings are being used properly under the given context, they compared their distributions across all pairs of semantic types (T_d, T_r) by using the

Kullback-Leibler divergence.

Based on the output of TextRunner [8], Kok and Domingos [66] built a Semantic Network Extractor (SNE) to detect groups of entities and relational phrases of relations. Their model, which is enhanced by two Markov Logic Networks (MLN), can simultaneously cluster both entities and phrases. The model learns the log-posterior of each cluster assignment Γ as shown in Equation 2.3, in which R is the set of relations, K is the set of cluster combinations, and t_k and f_k are the empirical numbers of true and false atom in a cluster combination k, respectively.

$$\log P(\Gamma|R) = \sum_{k \in K} \left[t_k \log \left(\frac{t_k + \alpha}{t_k + f_k + \alpha + \beta} \right) + f_k \log \left(\frac{f_k + \beta}{t_k + f_k + \alpha + \beta} \right) - \lambda m_{cc} + \mu d + C \right] \tag{2.3}$$

Other parameters in Equation 2.3 present the following meanings:

- $\alpha + \beta$: They are smoothing parameters used to estimate the MAP (maximum a posteriori) weight of an instance of the atom prediction rule.
- λ: A weight corresponds to the number of cluster combinations (m_{cc}) being formed.
- μ: A weight accompany with the number of pairs of symbols that belong to different clusters (d).

Yao et al. [158] addressed the problem of polysemy by disambiguating the sense of each cluster and using hierarchical agglomerative clustering to group them into semantic relations. To cluster the entity pairs of a single relation pattern into senses they employed a topic model that is modified based on latent Dirichlet allocation (LDA). In their model (Sense-LDA), they defined a path as a list of entity pairs co-occurring with the path in their tuples. The topic distribution of each path is drawn from a set of features, including entity names, words between and around two entities, document theme and sentence theme. Their experimental results showed that the approach discovered more accurate clusters than other baselines.

These above-mentioned models are unsupervised in the sense that no manual labeling of clusters by human is needed. One of the major shortcomings of their approaches, however, is that they only focus on the textual surface of arguments of a relation to estimate the synonymy probability and cannot effectively capture other features, such as the context around the relations. In this dissertation, we propose a representation for relational phrases that can capture both of the textual surface property and the context around them.

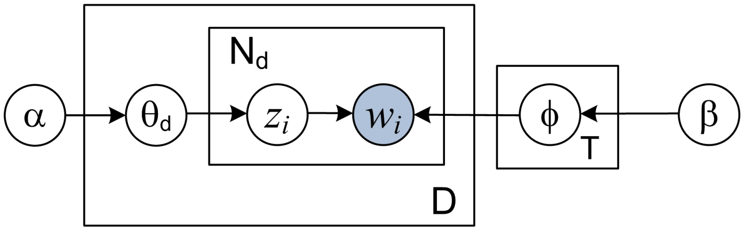


Figure 2.5: Graphical model representation of LDA. The boxes are plates representing replicates. The arrows indicate the conditional dependencies between two variables.

2.6 Latent Dirichlet Allocation

2.6.1 Formal Definition

Latent Dirichlet Allocation (LDA) is a generative model that provides full generative probabilistic semantics for documents [15]. Documents are modeled via a hidden Dirichlet random variable that specifies a probability distribution on a low-dimensional topic space. The distribution over words of an unseen document is a continuous mixture over document space and a discrete mixture over all possible topics.

Blei et al. [15] assumed that there are T underlying latent topics according to which documents are generated, and that each topic is represented as a multinomial distribution over $|W|$ words in the vocabulary. A document is generated by sampling a mixture of these topics and then sampling words from that mixture. The graphical model representation of LDA [15] is shown in Figure 2.5, in which, w_i is a word in a document, a topic $z_i \in \{1, ..., T\}$ is drawn from a Multi(θ) distribution $p(z_i = j|\theta) = \theta_j$, θ_d is sampled from a Dirichlet $(\alpha_1, ..., \alpha_T)$ distribution. The probability of the i word in a given document is calculated as follows [45]:

$$P(w_i) = \sum_{j=1}^{T} P(w_i|z_i = j)P(z_i = j) \tag{2.4}$$

We can represent $P(w|z)$ with a set of T multinomial distributions ϕ over the W words, such that $P(w|z = j) = \phi_w^{(j)}$.

Formally, LDA generates each word in the documents as follows [122]:

```
for each topic $t = 1 \ldots T$ do
  Generate $\phi_t$ according to symmetric Dirichlet distribution Dir($\beta$).
end for
for each document $d = 1 \ldots D$ do
  Generate $\theta_d$ according to Dirichlet distribution Dir($\alpha$).
```

```
    for each word $i = 1 \ldots W$ do
        Generate $z_{d,i}$ from Multinomial($\theta_d$).
        Generate the word $w_{d,i}$ from multinomial $\phi_{z_{d,i}}$.
    end for
end for
```

2.6.2 Gibbs Sampling for LDA

To estimate the parameters ϕ and θ, Griffiths and Steyvers [45] considered the posterior distribution over the assignments of words to topics, $P(\mathbf{z}|\mathbf{w})$. The complete probability model is

$$\begin{array}{lcl} w_i|z_i, \phi^{(z_i)} & \sim & \text{Discrete}(\phi^{(z_i)}) \\ \phi & \sim & \text{Dirichlet}(\beta) \\ z_i|\theta^{(d_i)} & \sim & \text{Discrete}(\theta^{(d_i)}) \\ \theta & \sim & \text{Dirichlet}(\alpha) \end{array}$$

in which α and β are assumed to be single values. Because $P(\mathbf{w}, \mathbf{z}) = P(\mathbf{w}|\mathbf{z})P(\mathbf{z})$ and ϕ and θ only appear in the first and second terms, respectively, we can perform integrals separately. When integrating out ϕ for the first term, we have

$$P(\mathbf{w}|\mathbf{z}) = \left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\right)^T \prod_{j=1}^{T} \frac{\prod_w \Gamma(n_j^{(w)} + \beta)}{\Gamma(n_j^{(.)} + W\beta)}, \tag{2.5}$$

in which $n_j^{(w)}$ is the number of times word w has been assigned to topic j, and $\Gamma(.)$ is the standard gamma function. Similarly, integrating θ for the second term, the result is

$$P(\mathbf{z}) = \left(\frac{\Gamma(T\alpha)}{\Gamma(\alpha)^T}\right)^D \prod_{d=1}^{D} \frac{\prod_j \Gamma(n_j^{(d)} + \alpha)}{\Gamma(n_{(d)}^{\cdot} + T\alpha)}, \tag{2.6}$$

The goal is to estimate the posterior distribution

$$P(\mathbf{z}|\mathbf{w}) = \frac{P(\mathbf{w}, \mathbf{z})}{\sum_z P(\mathbf{w}, \mathbf{z})} \tag{2.7}$$

However, we cannot directly compute this distribution because the sum in the denominator does not factorize and involves T^W terms. To tackle the problem, Griffiths and Steyvers used Gibbs sampling, where the next state is reached by sequentially sampling all variables from their distribution when conditioned on the current values of all other variables and data

[45]. To apply Gibbs sampling, it is necessary to calculate the full conditional distribution $P(z_i|\mathbf{z}_{-i}, \mathbf{w})$. Based on Equations 2.5 and 2.6, this distribution can be estimated as:

$$P(z_i = j|\mathbf{z}_{-i}, \mathbf{w}) \propto \frac{n^{(w_i)}_{-i,j} + \beta}{n^{(\cdot)}_{-i,j} + W\beta} \frac{n^{(d_i)}_{-i,j} + \alpha}{n^{(d_i)}_{-i,\cdot} + T\alpha}, \tag{2.8}$$

where $n^{(\cdot)}_{-i}$ is a count that exclude the current assignment of z_i. Equation 2.8 can be interpreted as the probability distribution approximates the probability of w_i under topic j (the first ratio) multiplied by the probability of topic j in the document d_i (the second ratio).

Having the full conditional distribution, the Gibbs sampler can be implemented as follows. First, the z_i variables are initialized to values in $\{1, 2, \dots T\}$. Then, by using Equation 2.8, we assign words to topics with counts that are computed from the subset of the words which have been seen so far rather than the full data. We run this step for a number of iterations; in each iteration a new state is found by sampling z_i. After enough iterations, the current values of z_i are recorded.

For any single sample, we can estimate ϕ and θ by

$$\hat{\phi}^{(w)}_j = \frac{n^{(w)}_j + \beta}{n^{(\cdot)}_j + W\beta} \tag{2.9}$$

$$\hat{\theta}^{(w)}_j = \frac{n^{(d)}_j + \alpha}{n^{(d)}_{\cdot} + T\alpha} \tag{2.10}$$

2.6.3 Applications

LDA is widely used in many NLP applications, e.g., relation extraction [70, 149, 157]. Yao et al. [157] proposed Rel-LDA and Type-LDA models for modeling tuples. In the Rel-LDA model, a set of relation tuples is considered as a *document*. Each relation tuple is represented by a set of its features, and the feature is generated independently from a hidden variable. The output of this model is a clustering of observed relation tuples. In order to capture the relationship between two arguments in a relation, they extended the Rel-LDA model to the Type-LDA model. In this model, in addition to a set of features as Rel-LDA, a relation is presented by two hidden entities types of the two arguments. These hidden types are drawn independently from the hidden relation. Each type is presented by a set of entity level features. Other extend versions of LDA have also been proposed to tackle relation extraction, e.g., ERD-MedLDA [70] and BioLDA [149].

LDA has been applied to selectional preference discovery [122, 126]. Ritter et al. [122] presented LDA-SP, which ultilizes the LinkLDA model [34], to extract inference rules between relation phrases. The model tries to capture the information about the pair of topics of the two arguments in a relation. In this model, each relation phrase is treated as a *document* and two sets of entities that share the same relation phrase are treated as *words* in the document. The experimental results on three different tasks (pseudo-disambiguation, selectional preferences for inference rules, and class-based selectional preferences) have shown the effectiveness of the model.

Séaghdha [126] also applied LDA models to selectional preferences. They proposed two models, namely ROOTH-LDA and DUAL-LDA, for this task. Unlike LDA-SP that models predicates with two arguments, in this study, they focused on predicates that take only a single argument, including verb-object, noun-noun and adjective-noun. ROOTH-LDA models the joint probability of co-occurring of a predicate and an argument, while DUAL-LDA models the classes of predicates and arguments separately. The two models performed competitively on identifying semantic classes on text corpus.

Other applications of topic models are sense disambiguation (Sense-LDA [158]), unsupervised coreference resolution [47], summarization [30], document alignment and segmentation [19].

2.7 Distributed Word Representations

2.7.1 Neural Network Language Model

Bengio et al. [9] proposed a neural network language model (NNLM) that can learn distributed representations for words based on their neighboring context. Given a sequence of $w_1 w_2 ... w_T$ of words, $w_t \in V$ and V is the vocabulary of the training data, the objective of the model is to maximize a function $f(w_t, ..., w_{t-n+1}) = \hat{P}(w_t | w_1^{t-1})$. The function is decomposed into two parts:

- A mapping C from any element i of V to a real vector $C(i) \in \mathbb{R}^m$, where C is a matrix of $|V| \times m$ and shared across all the words in the context.

- A function g that maps the input sequence $(C(w_{t-n+1}), ..., C(w_{t-1}))$ to conditional probability distributions of $P(w_t = i | C(w_{t-n+1}), ..., C(w_{t-1}))$.

As illustrated in Figure 2.6, the function g is implemented by a feed-forward neural network that consists of four layers: one projection layer, two hidden layers and one output layer. At the projection layer, each word in the input sequence is represented by a column in the matrix C. The column is indexed by the position of the word in the vocabulary. The concatenation or sum of the vectors is used as input word features x for the hidden layers.

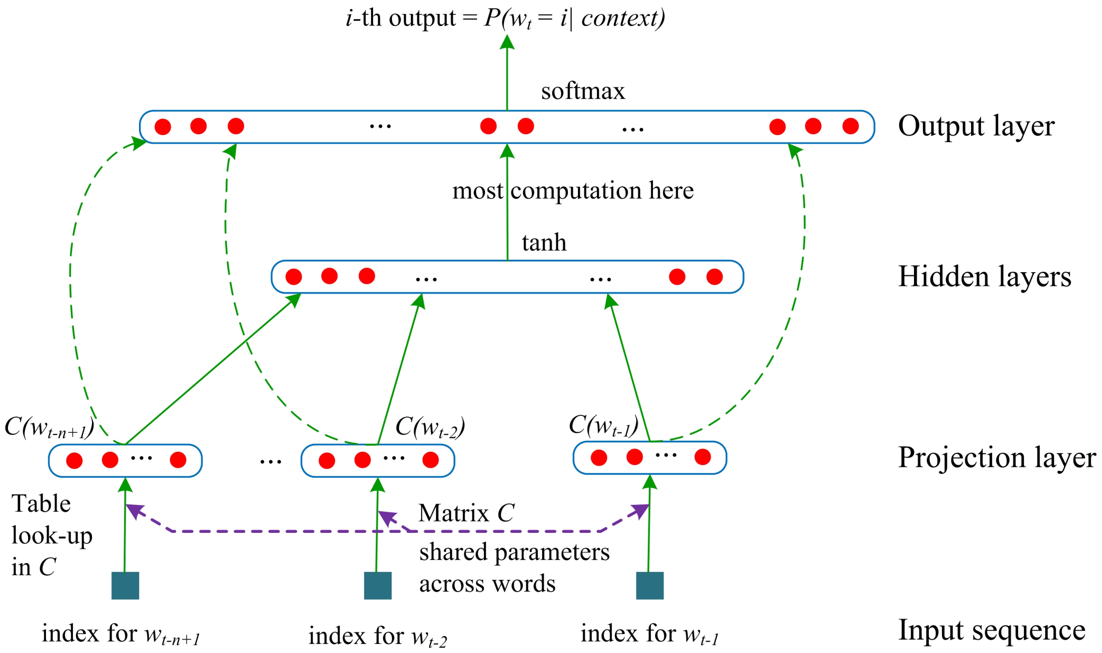


Figure 2.6: Neural architecture: $f(i, w_{t-1}, ..., w_{t-n+1}) = g(i, C(w_{t-1}), ..., C(W_{t-n+1}))$, where g is the neural network and $C(i)$ is the i-the word feature vector. [9]

Among the two hidden layers, there is an optional layer that directly connects from the projection to the output layer. The other layer, namely the ordinary hyperbolic tangent hidden layer, calculates unnormalized log-probabilities for each input word i as follows:

$$y = b + Wx + U\tanh(d + Hx), \tag{2.11}$$

where b, W, U, d and H are parameters that trained by using stochastic gradient ascent on the neural network.

Finally, the output layer computes the conditional probability through a $softmax$ function:

$$\hat{P}(w_t|w_{t-1}, ..., w_{t-n+1}) = \frac{e^{y_{w_t}}}{\sum_i e^{y_i}} \tag{2.12}$$

Interestingly, by using word distributed representations, the similarity of word can be computed by algebraic operations, e.g., C(‘King’) - C(‘Man’) + C(‘Woman’) is close to the vector of the word ‘Queen’ [80].

2.7.2 Hierarchical Softmax

Performing the normalization as shown in Equation 2.12 requires huge computation on the whole vocabulary V. To reduce the computational cost, hierarchical softmax is proposed

[91, 93].

The hierarchical softmax represents the output layers by a binary tree whose leaf nodes correspond to the V words and inner nodes include the relative probabilities of their child nodes. This setup replaces one V-way choice by a sequence of $\mathcal{O}(\log V)$ binary choice [91]. As a result, when we traverse from the root to the target word, we perform a sequence of $\mathcal{O}(\log V)$ local binary normalizations.

The concrete formula of the hierarchical softmax is defined by Mikolov et al. [82] as follows. Let $n(w, j)$ be the j-th node on the path from the root to w, and let $L(w)$ be the length of the path, so $n(w, 1) = \mathit{root}$ and $n(w, L(w)) = w$. For any inner node n, let $\mathrm{ch}(n)$ be an arbitrary fixed child of n and let $[\![x]\!]$ be 1 if x is true and -1 otherwise. Then the hierarchical softmax is computed as:

$$p(w|w_I) = \prod_{j=1}^{L(w)-1} \sigma\left([\![n(w, j+1) = \mathrm{ch}(n(w, j))]\!] \cdot {v'_{n(w,j)}}^{\top} v_{w_I}\right) \tag{2.13}$$

where $\sigma(x) = 1/(1 + \exp(-x))$, v_{w_I} is the vector representation of the context around the word w, and $v'_{n(w,j)}$ is the vector representation of the jth ancestor. Since $\sigma(x) + \sigma(-x) = 1$, it is easy to verify that $\sum_{w=1}^{V} p(w|w_I) = 1$.

The structure of the tree used by the hierarchical softmax affects considerably the training speed. Mnih and Hinton [91] proposed some methods for automatically constructing the tree structure and showed the effects on both of the training time and the accuracy of the model. Mikolov et al. [82] used a binary Huffman tree because the tree assigns short codes to the frequent words, which results in fast training.

2.7.3 Applications

Recently, distributed representations have been shown to effectively improve the performance of many NLP tasks.

Socher et al. [131] proposed unsupervised recursive autoencoders (RAE) to address the task of paraphrase detection. Given a list of word vectors produced by a NNLM, the autoencoders learn feature vectors for phrases in syntactic trees. They then compute a similarity matrix of two sentences based on the feature vectors. A softmax classifier is finally used to decide whether the two sentences are paraphrase or not. The experimental results showed that the autoencoders achieved state-of-the-art performance compared to previous approaches.

The authors then modified the autoencoders so that it can jointly learn phrase representations, phrase structure and sentiment distributions [132]. In order to predict a sentence-

or phrase-level target distribution, they extended RAE to semi-supervised setting by adding a supervised softmax layer on top of nonterminal nodes. The model then used L-BFGS to learn its parameters. They evaluated the model on standard datasets and the results showed that RAE outperformed competitive baselines.

Pyysalo et al. [107] reported that using word representations are beneficial for entity mention tagging in the biomedical domain. They first employed the Skip-gram model [82] to learn word vectors on all PubMed Central Open Access (PMC OA) texts. Next, they clustered the words to 1000 groups by using the *k*-means algorithm. Features for the supervised entity tagging were derived from the word clusters. Their evaluation on three biomedical corpora including the BioCreative II Gene Mention task, the Anatomical Entity Mention, and the NCBI Disease, revealed that their approach surpassed previous methods in two out of the three corpora.

Word representations have also been applied to machine translation. Out-Of-Vocabulary (OOV) is one of the most difficult problems of classical Statistical Machine Translation (SMT). However, Mikolov et al. [81] found that it is possible to tackle the problem by using distributed representations of words and phrases. The key idea behind their method is that when we represent words by using vector format, words that share the same concept in two different languages can have similar geometric arrangements in both spaces. Hence, we can learn an accurate linear mapping, namely a rotation and scaling, from one space to another. By using the linear mapping, the model can infer missing words based on other words in the vocabulary. The results on translating English to Spain indicated that the model could achieve a very high Precision@5 metric (around 90%).

Zou et al. [166] applied word representations to phrase-based SMT in another way. Given two sets of word representations in two languages and their alignment, they defined a new objective function that embodies both monolingual semantics and bilingual translation equivalence. As a result, the proposed neural model can learn bilingual semantic embedding for words across the two languages. The bilingual embeddings were then applied to a phrase-based SMT system and the experimental results showed that the model improved a BLEU score of 0.48 on NIST08 Chinese-English task.

Other applications of word representations are semantic relation classification [49], word alignment [137], and information retrieval [55].

Chapter 3

Binary Relation Extraction for Biomedical Texts

We have built a system, hereafter called PASMED, to extract binary relations from biomedical literature. The system uses a set of PAS patterns to detect candidates of semantic relations. First, Mogura [72], a high-speed version of the Enju parser [90], is employed to extract NP pairs that satisfy predefined PAS patterns from sentences. Next, named entities in the NP pairs are identified by MetaMap [6]. Because MetaMap uses string matching to map biomedical texts to the concepts in the UMLS Metathesaurus [16], its output contains many spurious entities. In order to remove false positives, we conduct post-processing using information on parts-of-speech and frequencies of entities. Finally, a relation between two entities is extracted if and only if the pair of semantic types is included in the UMLS Semantic Network[1]. An illustration of the working flow of our system is shown in Figure 3.1.

3.1 Predicate-Argument Structure Patterns

Since we attempt to extract unrestricted types of relations, there are no labeled corpora suitable for learning an extraction model. We therefore took a practical approach of creating PAS-based extraction patterns manually by observing actual linguistic expressions. We decided to use PASs in this work primarily because PASs are a viable formalism for building shallow semantic representations of biomedical verbs [27]. As a result of recent advances in parsing technology, there are now publicly available deep parsers that can output PASs and are both scalable and accurate. The Enju parser is one of those parsers and has shown to

[1] The **U**nified **M**edical **L**anguage **S**ystem Semantic Network (http://semanticnetwork.nlm.nih.gov/)

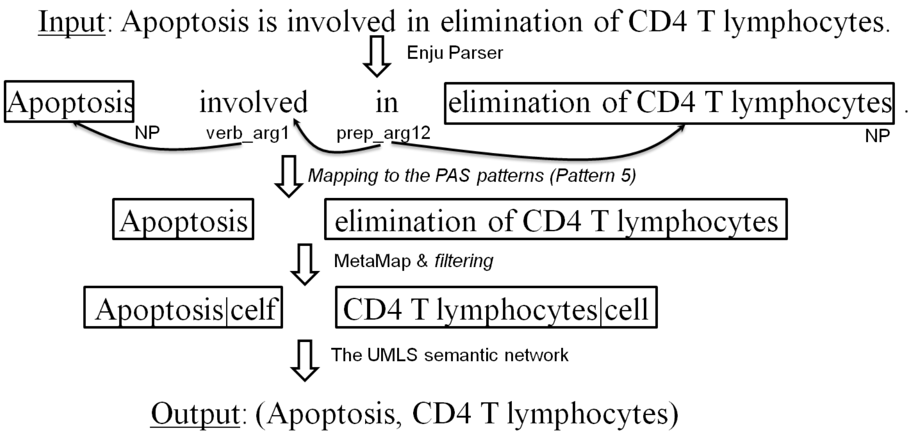

Input: Apoptosis is involved in elimination of CD4 T lymphocytes.
Enju Parser
Apoptosis
involved
in
elimination of CD4 T lymphocytes
NP
verb_arg1
prep_arg12
NP
Mapping to the PAS patterns (Pattern 5)
Apoptosis
elimination of CD4 T lymphocytes
MetaMap & filtering
Apoptosis|celf
CD4 T lymphocytes|cell
The UMLS semantic network
Output: (Apoptosis, CD4 T lymphocytes)

Figure 3.1: The working flow of our system.

be one of the most accurate syntactic parsers for biomedical documents [90].

3.1.1 Crafting Patterns

In order to find appropriate PAS patterns, we have first observed textual expressions that represent biomedical relations in the GENIA corpus [61] and found that those relations are usually expressed with verbs and prepositions. Examples of those are $Entity_A$ {*affect, cause, express, inhibit ...*} $Entity_B$ and $Entity_A$ {*arise, happen, ...*} {*in, at, on ...*} *Location*. Based on these observations, we create patterns that consist of three elements: (1) NP_1 containing $Entity_A$, (2) NP_2 containing $Entity_B$ and (3) a verbal or prepositional predicate that has the two NPs as arguments. Our patterns in predicate-argument form and their corresponding examples are presented in Table 3.1. It should be noted that no sentences in the GENIA corpus, which we examined for developing these patterns, were used in the evaluation experiments described in Section 3.3.

Pattern 1 and 2 capture expressions of transitive verbs in active and passive voices respectively. Their relevant NP pairs consist of subjects and objects of the verbs. Pattern 3 deals with verbal structures in which transitive verbs modify a noun phrase to describe specific actions, e.g., 'play a role' and 'produce changes'. Pattern 4 is used for linking verbs. A linking verb modifies an adjective. Hence, if a noun phrase related to the adjective is found, the phrase and the subject of the verb form a relevant NP pair. To deal with intransitive verbs, we use Pattern 5. An intransitive verb has no direct object, but it can be modified by a prepositional phrase to describe in detail about the action. In this case, the prepositional phrase and the subject of the verb constitute a relevant NP pair. The final pattern (Pattern

Table 3.1: Our PAS patterns focus on verb and preposition predicates. An arrow going from a to b means that a modifies b, where a is called a predicate, and b is called an argument. $<NP_1, NP_2>$ is a relevant NP pair in each pattern.

No.	PAS Patterns	Examples
1	$NP_1 \leftarrow \textbf{Verb} \rightarrow NP_2$	protein RepA(cop) ← affects → a single amino acid
2	$NP_1 \leftarrow \textbf{Verb} \rightarrow by + NP_2$	Diabetes ← induced → by streptozotocin injection
3	$NP_1 \leftarrow \textbf{Verb} \rightarrow NP'$ $\uparrow$ $Prep. \rightarrow NP_2$	Endothelin-1 (ET-1) ← had → a strong effect ↑ *in* → all trabeculae
4	$NP_1 \leftarrow \textbf{Link.Verb} \rightarrow ADJP$ $\uparrow$ $Prep. \rightarrow NP_2$	EPO receptor ← be → present ↑ *in* → epithelial cells
5	$NP_1 \leftarrow \textbf{Verb} \leftarrow Prep. \rightarrow NP_2$	Apoptosis ← involved ← in → CD4 T lymphocytes
6	$NP_1 \leftarrow \textbf{Prep.} \rightarrow NP_2$	vitronectin ← in → the connective tissue

6) is used for prepositions, which would capture localization and whole-part relations.

It should be noted that although the first three patterns for transitive verbs seem to overlap each other, they do capture different instances. When we map Enju's output to these patterns, if the input contains a transitive verb, it can only satisfy one pattern.

The elements NP_1 and NP_2 in each pattern shown in Table 3.1 are considered as candidates of our relation extraction step. For instance, the resulting candidate from the first example is (protein RepA(cop), a single amino acid), and so on for the other ones.

In order to estimate the coverage of our patterns, we applied them to three protein-protein interaction (PPI) corpora (AIMed, BioInfer and LLL [1, 106]), two drug-drug interaction (DDI) corpora (MedLine and DrugBank [127]), and the GENIA corpus [61]. We then checked if the entities in the annotated relations are included in the NP pairs of our patterns. For instance, according to the AIMed corpus, there is a PPI between 'IFN-gamma' and 'IFN-alpha' in the sentence "Levels of IFN-gamma is slightly increased following IFN-alpha treatment". This PPI is covered by Pattern 2, in which NP_1 is 'Levels of IFN-gamma' and NP_2 is 'IFN-alpha treatment'.

The results in Table 3.2 show that the patterns cover over 80% of the entities in the GENIA events and PPIs of the LLL corpus sufficiently. This is somewhat expected since our PAS patterns are created based on the observations on the GENIA corpus and the LLL corpus contains only 50 sentences. However, for the other cases, our patterns only cover a small portion, e.g., 46% relations of the BioInfer, and 53% of the AIMed. Relations that our patterns miss can be categorized into two groups: (1) nominal relations, e.g., 'CD30/CD30L interaction', and (2) relations that need other information, such as coreference resolution,

Table 3.2: Expected recall of our PAS patterns on various corpora.

PPI			DDI		GENIA
AIMed	BioInfer	LLL	MedLine	DrugBank	
53%	46%	82%	64%	62%	80%

to be inferred. These kinds of relations are hard to identify by only using a pattern-based approach and are left for future work.

3.1.2 Matching Patterns

Based on the Mogura's output format, we encoded our PAS patterns by using data structures as follows:

```
structure PASPattern
{
    string pred_type;
    PASComp *comp;
};
```

```
structure PASComp
{
    string pred_type;
    PASComp arg1, arg2;
};
```

To extract candidates of relations, we input the PAS of each sentence (by the Mogura parser) to our matching algorithm shown in Algorithm 1. The algorithm receives the PAS of a sentence and our PAS patterns as input, and outputs a set of tuples containing two noun phrases and a relation phrase. This is an exhaustive search algorithm that traverses all words in a sentence and match them with the patterns. If there is a word whose predicate type is identical to that of a PAS pattern, the algorithm will match the word's argument with the first component of the pattern to detect the first noun phrase. Then it will match other components of the pattern with other words until it finds the second noun phrase and the relation phrase, or there is no word or component left.

The complexity of this searching algorithm is $\mathcal{O}(n)$ in the best case when there is no word satisfying the predicate type of the PAS patterns. The complexity in the worst case is $\mathcal{O}(n^2)$ when the algorithm has to traverse $(n-1)$ other words to find np_2.

3.2 Extracting Semantic Relations

After obtaining NP pairs by matching sentences to the patterns, our system relies on MetaMap for named entity recognition and post-processing identified relations by check-

Algorithm 1 The brute-force algorithm to match our PAS patterns.

```
Input:
    PAS of a sentence with $n$ words, $W = \{w_1, w_2, ..., w_n\}$
    PAS patterns, $Patt = \{pa_1, pa_2, pa_3, pa_4, pa_5, pa_6\}$
Output: tuples $T = \{< np_{11}, rp_1, np_{12} >, ..., < np_{m1}, rp_m, np_{m2} >\}$
for each $w_i \in W$ do
    for each $pa_j \in Patt$ do
        if predicate type of $w_i$ equals to $pa_j.pred\_type$ then
            if predicate type of argument 1 of $w_i$ equals to $pa_j.comp[0].pred\_type$ then
                $np_1$ := argument 1 of $w_i$
            else
                continue
            $rp := w_i$
            for each $w_{-i} \in W$ do
                find $np_2$ and update $rp$ by matching $w_{-i}$ with $pa_j.comp$
            insert $< np_1, rp, np_2 >$ to $T$
return $T$
```

ing semantic types in the UMLS Semantic Network.

3.2.1 Named Entity Recognition

Named entity recognition (NER) is an important text processing step that needs to be performed before relation extraction. Most of previous machine-learning NER tools have focused on detecting gene/protein names [64], gene/protein, cell line and cell type [62], drugs and chemicals [147]. Those tools perform well with the targeted entities but it is not easy to extend them to other types of entities. Moreover, they only locate entities in text and do not offer other information such as global identifiers (IDs) of the recognized entities, which will be useful for linking them with information stored in biomedical databases. In this work, we use MetaMap [6], a dictionary-based tool that maps biomedical texts to the concepts in the UMLS Metathesaurus [16].

The Metathesaurus is a large database that contains biomedical and clinical concepts from over 100 disparate terminology sources. In order to integrate them into a single resource, a unique and permanent concept identifier (CUI) is assigned to synonymous concepts or

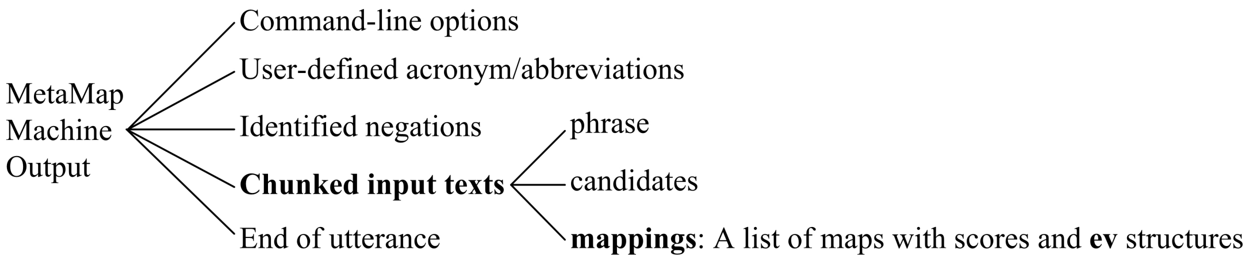


Figure 3.2: A hierarchical structure of MMO.

meanings[2]. For instance, the Metathesaurus maps the two strings of 'Chronic Obstructive Lung Disease' from Medical Subject Headings (MSH) and 'COLD' from National Cancer Institute thesaurus (NCI) to a concept whose CUI is 'C0009264'. By using MetaMap, we can obtain the CUI and the source names of an entity. Although MetaMap does not perform as well as machine-learning tools in terms of recognition accuracy, it meets our requirement of detecting every entity in texts and outputs the Metathesaurus concept unique identifier (CUI), i.e., a global ID for each entity.

Reading MetaMap's Output

MetaMap delivers its output for the whole MEDLINE[3] in the form of MetaMap Machine Output (MMO)[4]. MMO consists of five main kinds of objects as shown in Figure 3.2. These objects are always placed in a fixed order.

Among the five objects, we focus on the chunked input texts, namely *utterance* objects, to extract information about detected entities. More specifically, we extract the list of *mappings* in each *utterance*. Each mapping is presented by a negative score and an *ev* structure with twelve fields: the score of the candidate mapping, CUI, the matched string, the preferred name of the concept, the lowercased words, the semantic types of the concept, the matched mapping, the head of the matched phrase, the overmatch field, the list of unique sources in which the concept appears, the position of the matched phrase, and the status of the concept. A list of mappings of the phrase 'Eye' is presented in Table 3.3.

It should be noted that for each identified entity, there are occasionally more than one mapping. In such cases, we select the mapping that has the highest score. When all mappings have the same score, we choose the first one. For instance, there are two mappings for the phrase 'Eye' in the example in Table 3.3, the first mapping will be selected. The obtained

[2] http://www.ncbi.nlm.nih.gov/books/NBK9676/

[3] http://mbr.nlm.nih.gov/Download/MetaMapped_Medline/2012/

[4] http://metamap.nlm.nih.gov/Docs/2012_MMO.pdf

Table 3.3: An example of MetaMap Machine Output with the *utterance* object.

```
utterance(‘16691646.ti.1’,“Statement of Cases of Gonorrhoeal and Purulent Ophthalmia
treated in the Desmarres (U. S. Army) Eye and Ear Hospital, Chicago, Illinois with
Special Report of Treatment Employed.”,156/191,[228,303]).
...
phrase(‘Eye’,[head([lexmatch([eye]),inputmatch([‘Eye’]),tag(noun),tokens([eye])])],
258/3,[]).
mappings([
map(-1000,[ev(-1000,‘C0015392’,‘Eye’,‘Eye’,[eye],[bpoc],[[[1,1],[1,1],0]],yes,no,[‘COSTAR’,
‘FMA’,‘HL7V2.5’,‘LCH’,‘LNC’,‘MSH’,‘MTH’,‘NCI’,‘SNM’,‘SNOMEDCT’,‘UWDA’,
‘AOD’,‘CHV’,‘CSP’,‘ICPC’,‘OMIM’,‘SNMI’,‘ICF-CY’,‘ICF’],[258/3],0)]),
map(-1000,[ev(-1000,‘C1280202’,‘Eye’,‘Entire eye’,[eye],[bpoc],[[[1,1],[1,1],0]],yes,no,
[‘MTH’,‘SNOMEDCT’],[258/3],0)])
]).
...
‘EOU’.
```

mapping can be interpreted as ‘Eye’ is an entity that has a concept identifier of ‘C0015392’, a semantic type of ‘Body Part, Organ, or Organ Component’ (bpoc); this entity appears in many resources, e.g., MeSH (‘MSH’) and Online Mendelian Inheritance in Man (‘OMIM’); and its position in the original text is 258 with a length of 3 characters.

Post-processing Entities

Since MetaMap uses string matching techniques to identify entities, it generates many false positive entities. We apply two post-process steps to remove these entities from MetaMap's output. In the first step, we remove all entities that are verbs, adjectives, prepositions or numbers because we are only interested in noun or noun phrase entities. The second step is used to avoid common noun entities, e.g., ‘study’, ‘result’ and ‘relative’. We first construct a dictionary of named entities based on MetaMap's results of the whole MEDLINE and remove highly frequent entities from it. This dictionary is then used to check the validity of named entities.

To evaluate the effectiveness of these post-processing steps, we conducted a small set of experiments using several annotated corpora. We employed MetaMap to detect proteins in

Table 3.4: Performance of our post-processing on proteins and drugs detection. These scores were generated by using the CoNLL 2000 evaluation script.

Protein	**Acc.**	**Pre.**	**Re.**	**F. (%)**
MetaMap	58.10	15.72	63.21	25.18
After filtering	***88.93***	***55.77***	47.61	***51.37***
Drug				
MetaMap	62.61	20.86	79.51	33.04
After filtering	***93.96***	***83.26***	62.47	***71.38***

AIMed, BioInfer and LLL [1, 106], and drugs in the SemEval-2013 task 9 corpus [127]. We then post-processed these outputs and compared them with labeled entities to evaluate the performance of our post-processing. The scores in Table 3.4 show that our filtering improved the F-scores significantly for both proteins and drugs, resulting in F-scores of 51.37% on proteins and 71.38% on drugs. This performance is comparable to that of CubNER, an unsupervised NER tool for biomedical text [162].

Moreover, our statistical results on the whole MEDLINE show that the post-processes filtered about **70.83%** entities out of the MetaMap output. This filtering helps our system avoid extracting irrelevant relations.

3.2.2 Relation Extraction

We obtain named entities in candidates of NP pairs after our post-processes. Next, each entity in NP_1 is coupled with every entity in NP_2 to create a candidate of semantic relation. It should be noted that separate entities inside a noun phrase are not considered to constitute a relation. Let us denote by $<NP_1, NP_2>$ a relevant NP pair, by e_{1i} $(i = 1, 2, ...)$ entities in NP_1, and by e_{2j} $(j = 1, 2, ...)$ entities in NP_2. Every pair of entities $<e_{1i}, e_{2j}>$ can compose a candidate of semantic relation. However, this process may create many spurious relations. Therefore, we use the UMLS Semantic Network as a constraint of extracting semantic relations to improve the precision of our system.

The Semantic Network consists of (1) a set of 133 semantic types that provides a consistent categorization of all concepts represented in the UMLS Metathesaurus, and (2) a set of semantic relations that exists between semantic types. Therefore, to determine a correct relation we need to know the right semantic type of each entity. Although MetaMap involves

Table 3.5: Frequency of semantic types of *methyl violet* and *hydrochloride* in MEDLINE.

Entity	Semantic Type	Count
methyl violet	**Indicator, Reagent, or Diagnostic Aid**	**170**
	Pharmacologic Substance	170
	Organic Chemical	170
hydrochloride	**Inorganic Chemical**	**13265**
	Pharmacologic Substance	13259
	Amino Acid, Peptide or Protein	6
	Immunologic Factor	6
	Organic Chemical	5
	Element, Ion, or Isotope	4
	Antibiotic	2
	Classification	2
	Quantitative Concept	2
	Indicator, Reagent, or Diagnostic Aid	1

a word sense disambiguation process [6], it usually assigns multiple semantic types to a single entity. For instance, in the above example, MetaMap assigns 'Indicator, Reagent, or Diagnostic Aid' (irda) to *intercalators*; 'Indicator, Reagent, or Diagnostic Aid' (irda), 'Organic Chemical' (orch) and 'Pharmacologic Substance' (phsu) to *methyl violet*; and 'Inorganic Chemical' (inch) and 'Pharmacologic Substance' (phsu) to *hydrochloride.* In order to choose one semantic type for each entity, we calculate the frequency of each pair of entity-semantic type in the whole MEDLINE and assign the highest frequency semantic type to the entity. Table 3.5 shows the entity-semantic type pairs of *methyl violet* and *hydrochloride.* Our system assigns *Indicator, Reagent, or Diagnostic Aid* to *methyl violet*, and *Organic Chemical* to *hydrochloride* because those semantic types have higher frequencies than the others.

As mentioned above, the Semantic Network provides a relation ontology that consists of a set of relations between semantic types, such as relations between 'Gene or Genome' and 'Enzyme', or 'Hormone' and 'Disease or Symptom'. Let us denote by $<s_1, s_2>$ the pair of semantic types of $<e_{1i}, e_{2j}>$. If and only if $<s_1, s_2>$ exists in this relation ontology, $<e_{1i}, e_{2j}>$ can constitute a relation. For example, we obtained a candidate pair of (Diabetes, streptozotocin injection) from the second row in Table 3.1. The MetaMap's output of this pair is <Diabetes, streptozotocin> and its corresponding pair of semantic types is <Disease

or Symptom, Antibiotics>. According to the Semantic Network, there are relations between these two semantic types, such as 'affected_by', 'treated_by' and 'prevented_by'. Therefore, <Diabetes, streptozotocin> can constitute a semantic relation in our system.

3.3 Experiments and Results

We have conducted two series of experiments to evaluate the performance of our system on general and pre-defined relations. Regarding general relations, we have created a test set of 500 sentences by randomly selecting from MEDLINE and manually evaluated the output of our system according to our criteria presented below. By setting this evaluation, we attempt to estimate the performance of PASMED from a perspective of open-domain relation extraction from MEDLINE. While with pre-defined relations, we automatically evaluated our system on protein-protein and drug-drug interactions by using gold standard corpora. In both series, we have compared our system with other state-of-the-art systems.

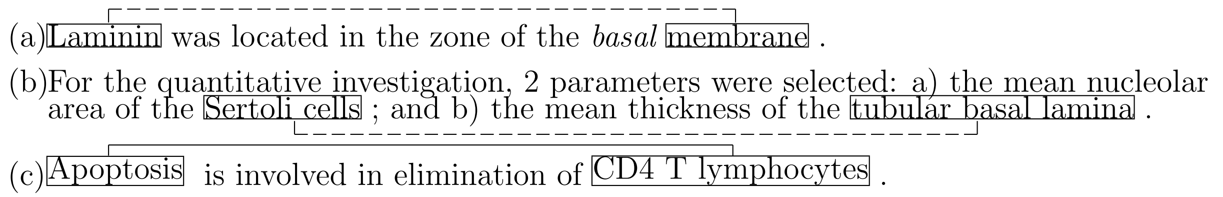
3.3.1 Criteria of Manual Evaluation

In this work, an extracted relation is a biomedical binary relation composed by two biomedical entities and it usually represents some association or effect between the entities. To evaluate these relations, we have defined evaluation criteria for entities and relations.

Evaluating Entities: An entity is correct if and only if (1) it is a noun or a base noun phrase (a unit noun phrase that does not include other noun phrases), and (2) its content words represent the complete meaning within the sentence containing it. The first condition is set up in this criterion because MetaMap can only detect entities that are nouns or base noun phrases. The second one is to guarantee the meaning of the annotated entities. For example, Figure 3.3(a) shows a relation between two entities 'Laminin' and 'membrane'. In this case, the entity 'Laminin' is correct, but the entity 'membrane' is not. The reason is that 'membrane' does not reflect the full meaning intended in this sentence; the right entity should be 'basal membrane'.

Evaluating Relations: A correct relation must satisfy the following two conditions:

- The two entities composing the relation must be correct according to the above-mentioned criterion.



(a) Laminin was located in the zone of the *basal* membrane .

(b) For the quantitative investigation, 2 parameters were selected: a) the mean nucleolar area of the Sertoli cells ; and b) the mean thickness of the tubular basal lamina .

(c) Apoptosis is involved in elimination of CD4 T lymphocytes .

Figure 3.3: Examples of biomedical binary relations. (a) The relation is not correct because of one incorrect entity. (b) The relation is not correct because the relationship between the two entities is not represented explicitly by any semantic clue. (c) The relation is correct because it satisfies our two criteria of manually evaluation.

- The relationship between two entities in a correct relation must be described *explicitly* by some linguistic expression.

Any relations that break one of the above conditions are considered to be incorrect. For example, the extracted relation in Figure 3.3(c) is correct since it meets our criteria, while the extracted relations in (a) and (b) are not. The relation in (a) does not meet the first criterion since the entity ‘membrane’ is not correct. The relation in (b) does not meet the second criterion because this sentence only lists two selected parameters that are related to ‘Sertoli cells’ and ‘tubular basal lamina’, and no relationship between these two entities is mentioned. More details about our evaluation guideline can be seen in Appendix A.

3.3.2 Results of General Relations

For the purpose of evaluation, we have created our original test set from MEDLINE since there is no labeled corpora for evaluating the various types of relations targeted in this thesis. This test set was created by randomly selecting 500 sentences from MEDLINE. Our system was given this set as input, and returned a set of binary relations as output. For comparison, we conducted experiments using two state-of-the-art Open IE systems for general domains, namely, ReVerb [35] and OLLIE [73]. We employed these two systems to extract relevant NP pairs in place of our PAS patterns. The other processes were applied in exactly the same way as our system. We also compared our system with the latest version of SemRep[5] on the test set.

Two annotators were involved in evaluating general relations. The two annotators have different backgrounds. Annotator A has a PhD in biology, majoring in genetics. Annotator B has a master degree of computer science, majoring in natural language processing; he is

[5] http://semrep.nlm.nih.gov/

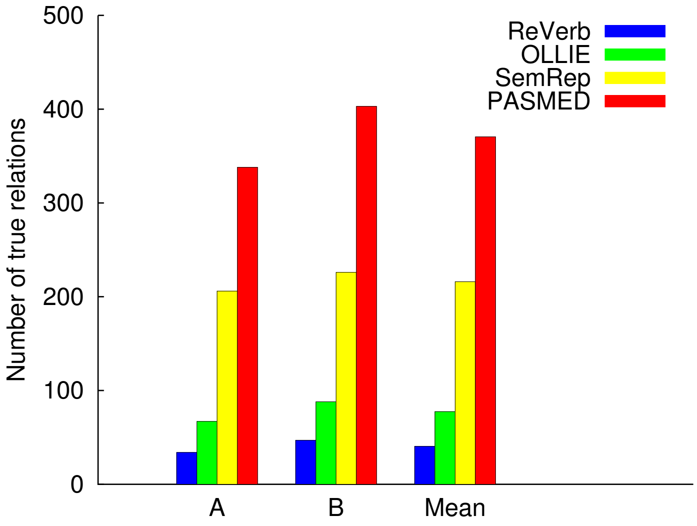


Figure 3.4: The number of true relations of the four systems on our test set according to the agreement of the two annotators. The mean numbers are 40.5, 77.5, 216, and 370.5, respectively. PASMED achieved the highest ones in all cases.

also a bachelor of medical biotechnology. The annotators were required to strictly follow our criteria when evaluating the outputs of the four systems: ReVerb, OLLIE, SemRep and PASMED. Both Annotator A and B were blind to the identity of the systems, i.e., they do not know which output was produced by which system.

Both ReVerb and OLLIE assign a confidence value to each extracted triple instead of simply classifying them as true or false. In our experiments, this value was used as the threshold for extracting relations. We selected the values generating the best harmonic mean of precision and the number of true positives in our experiments, which turned out to be 0.7 for both systems. On our test set, ReVerb, OLLIE, SemRep and PASMED extracted 77, 164, 346, and 781 relations, respectively.

Figure 3.4 shows the numbers of true relations output by the four systems according to the two annotators. PASMED identified the highest number of true relations than the other systems. Specifically, the number of true relations extracted by PASMED was 71% higher than that of SemRep, which was the second best among the four systems. It should be noted that we can decrease the thresholds of ReVerb and OLLIE to increase their recalls. However, even when the thresholds were 0.3, their numbers of true positive relations were much lower than that of PASMED, which were about 52 and 103 on average, respectively.

In order to estimate the recall of these systems, we used *relative* recall defined by Clarke

Table 3.6: Evaluation results of the four systems according to the two annotators. SemRep achieves the highest precision, PASMED achieves the highest relative recall.

System	Annotator A			Annotator B			Mean		
	Pre.	Re.	F.	Pre.	Re.	F.	Pre.	Re.	F.
ReVerb	44.15	6.75	11.72	61.04	9.34	16.20	52.59	8.05	13.96
OLLIE	40.85	13.32	20.10	53.65	17.49	26.38	47.25	15.41	23.24
SemRep	59.37	40.95	48.47	65.13	38.83	48.65	62.25	39.89	48.56
PASMED	43.27	67.19	52.65	51.50	69.24	59.13	47.39	68.22	55.89

and Willett [25]. Let *a, b, c* and *d* denote the true relations of ReVerb, OLLIE, SemRep and PASMED respectively. We created a pool of gold-standard relations by merging *a, b, c, d* and removing duplicates. Let r denote the number of relations in the pool ($a, b, c, d < r \leq a + b + c + d$), the recall of ReVerb is calculated as a/r and similarly for the other systems. We reported all scores of the four systems in Table 3.6. The higher recalls of PASMED in the table are in large part explained by the fact that the system has no restriction in predicate types, thereby accepting diverse biomedical relations. SemRep achieves a better precision score than PASMED by restricting the predicate types with its ontology but misses many relations due to the constraint. These results will be analyzed in more detail in the next section.

A significance test on the F-scores of SemRep and PASMED was conducted by using approximate randomization [100]. We performed 1000 shuffles on the output of SemRep and PASMED and the approximate p-values according to the two annotators A and B are 0.35 and 0.02, respectively. These p-values indicate that with a rejection level of 0.05, there is a chance that the difference between SemRep and PASMED is statistically significant, which can be interpreted as the overall performance of PASMED is better than SemRep.

We have also calculated the Inter-Annotator Agreement (IAA) in each system by using κ statistics adapted to multiple coders [38]. We reports the values and their scales according to Landis and Koch (1977) [68] and Green (1997) [44] in Table 3.7. The IAA scales indicate that the evaluation results are reliable enough.

3.3.3 Error Analysis

We have listed the numbers of PASMED's false positive relations caused by different types of errors in Table 3.8. On average, our system generated 410.5 false positive relations;

Table 3.7: The inter-annotator agreement scores of the four systems and their corresponding scale according to two different standards.

IAA	ReVerb	OLLIE	SemRep	PASMED
	0.664	0.598	0.680	0.741
Landis and Koch (1977)	Substantial	Moderate	Substantial	Substantial
Green (1997)	Good	Good	Good	Good

Table 3.8: Numbers of false positive PASMED's relations according to the two annotators. We have classified them into three types of errors: C1–false positives caused by incorrect entity extraction (criterion 1), C2–false positives caused by not presented explicitly by linguistic expressions (criterion 2), and Both–false positives due to both C1 and C2.

	C1	C2	Both	Total
Annotator A	257	120	66	443
Annotator B	311	50	17	378
Mean	284 69.18%	85 20.71%	41.5 10.11%	410.5

among them (1) about 69.18% of them (284 false positive ones) are due to incorrect entitiy extraction (criterion 1), (2) 20.71% of false positive ones are not presented explicitly by linguistic expression (criterion 2) and (3) 10.11% break both criteria. The reason for the first case is that MetaMap occasionally fails to capture named entities with multiple tokens like the example in Figure 3.3(a).

The second case is caused by parser errors and our greedy extraction. For instance, with this input:

> Laminin was *located in* the zone of the basal membrane, whereas tenascin was mainly found in the mucosal vessels.

The system detected a NP pair as follows

NP$_1$: [Laminin]

NP$_2$: the zone of the basal [membrane], *whereas* [tenascin] was mainly found in the mucosal [vessels]

Based on the NP pair, the system returned three relations: r_1 (Laminin, membrane), r_2 (Laminin, tenascin), and r_3 (Laminin, vessels). Among them, r_2 and r_3 break both evaluation conditions. In this example, the parser failed to detect the second NP of the pair; the correct one should be 'the zone of the basal membrane', not including 'whereas' clause. Then, from this incorrect pair, our greedy extraction generated r_2 and r_3 since we assume that every pair of entities in a NP pair constitutes a relation; even using the Semantic Network could not help in this case.

As reported in the previous section, PASMED extracted many more relations than the other three systems. In the case of ReVerb and OLLIE, the main reason for their low performance is that these systems failed to capture NP pairs in many sentences. More specifically, ReVerb and OLLIE could not extract NP pairs from 150 sentences and 95 sentences respectively; our system could not extract pairs only from 14 sentences. Given the input sentence:

> Total protein, lactate dehydrogenase (LDH), xanthine oxidase (XO), tumor necrosis factor (TNF), and interleukin 1 (IL-1) were *measured in* bronchoalveolar lavage fluid (BALF).

ReVerb and OLLIE could not extract any tuples, while our system generated a NP pair of

> NP_1: [Total protein], [lactate dehydrogenase] (LDH), [xanthine oxidase] (XO), [tumor necrosis factor] (TNF), and [interleukin 1] (IL-1)
>
> NP_2: [bronchoalveolar lavage fluid] (BALF)

and returned five correct relations between 'bronchoalveolar lavage fluid' and five entities in NP_1. In the case of SemRep, the main reason why it detected fewer relations than PASMED is that SemRep is restricted with a fixed set of verbs, which limits the set of relations found. For instance, SemRep also fails to extract relations in the above sentence because its ontology does not include the verb 'measure'.

However, since our PAS patterns focus on verbs and prepositions, there are relations that our system misses unlike SemRep, e.g., relations in the forms of modification/head of noun phrases. For example, SemRep identified a relation between 'tumor' and 'malignancy' in the sentence "Spontaneous [apoptosis] may play a role in evolution of [tumor] [malignancy]" while our system could not. It, instead, extracted the relation of ('apoptosis', 'malignancy') based on the phrase 'play a role in'.

Our system does not extract some relations that SemRep does since it filters MetaMap's output. Given the sentence "We monitored a group of [*patients*] with [pollinosis] sensitive to Olea.", SemRep output a relation between 'patients' and 'pollinosis'. PASMED ruled out 'patients' from MetaMap's output at its filtering step because this entity is an overly frequent entity in MEDLINE.

Nevertheless, this filtering step helps our system to discard many spurious relations that SemRep does not. For example, given the phrase "Morbidity risk for [alcoholism] and [drug abuse] in [*relatives*] of [cocaine addicts]", two relations ('relatives', 'alcoholism') and ('relatives', 'drug abuse') were extracted by SemRep. The two annotators assessed these relations as incorrect on the ground that the word 'relatives' alone is not specific enough. By contrast, PASMED discarded 'relatives' because this entity is too frequent in MEDLINE. No relation composed by the entity was thus identified. Instead, PASMED detected two other relations, ('alcoholism', 'cocaine addicts') and ('drug abuse', 'cocaine addicts'), which were assessed as correct by the annotators. We should note, however, that these relations are not strictly correct either, since the full description for the latter entity should be 'relatives of cocaine addicts'.

As for the set of PAS patterns used in PASMED, it is not impossible to extend them to detect more relations. The maximal recall that could be reached is about 80% in the best case (the same recall of the GENIA corpus, see Table 3.2), but there is a higher risk that the precision will be decreased substantially due to three sources of errors, including MetaMap's errors, parser's errors and our greedy extraction. Currently, PASMED relatively covers 68.22% of general relations on average, which we deem to be high enough for the current trade-off.

Here we clarify the differences—besides the fact that PASMED uses deep syntax—between ReVerb, OLLIE, SemRep and PASMED, which are all based on a pattern-based approach. Regarding ReVerb and OLLIE, a major difference is that they employ a parser for the general domain while PASMED uses a parser specifically tuned for the biomedical domain. One of the biggest differences between SemRep and PASMED is the way the extracted relations are verified. SemRep restricts its relations using a predefined predicate ontology based on the Semantic Network. PASMED also depends on the Semantic Network but uses it in a less restrictive manner, which contributed to the systems higher recall.

3.3.4 Evaluating on Predefined Relations

We also conducted experiments to see how well our PAS patterns cover predefined relations such as Protein-Protein Interaction (PPI) and Drug-Drug Interaction (DDI). Regarding PPI, we applied our patterns to AIMed, BioInfer and LLL–three popular corpora in this domain [1, 106]. The gold-standard entities available in these corpora were used instead of MetaMap output. We conducted the same experiment for DDI on the SemEval-2013 task 9 corpus [127].

For comparison and reference, we show the precision and recall of some notable systems on PPI and DDI. It should be noted that since these systems used machine learning methods,

Table 3.9: Performance of our system on AIMed, BioInfer and LLL corpora, compared with some state-of-the-art systems for PPI.

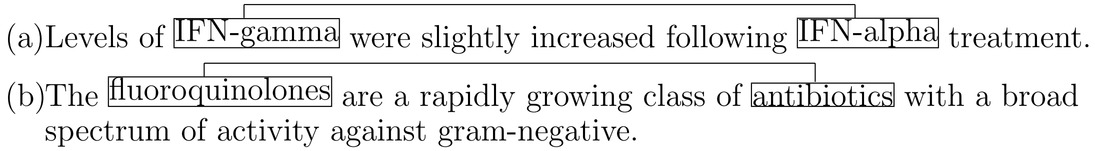
	AIMed		**BioInfer**		**LLL**	
	Pre.	Re.	Pre.	Re.	Pre.	Re.
Yakushiji et al.[155]	71.8	48.4	-	-	-	-
Airola et al.[1]	52.9	61.8	47.7	59.9	72.5	87.2
Miwa et al.[87]	55.0	68.8	65.7	71.1	77.6	86.0
PASMED	33.9	58.7	53.9	47.3	87.2	81.2

Table 3.10: Performance of our system on MedLine and DrugBank corpora of SemEval-2013 Task 9 [127], compared with the highest and lowest-performing systems in that shared task.

	MedLine		**DrugBank**	
	Pre.	Re.	Pre.	Re.
Highest-performing system	55.8	50.5	81.6	83.8
Lowest-performing system	62.5	42.1	38.7	73.9
PASMED	27.0	62.5	41.0	61.6

they were evaluated by using 10-fold cross-validation or using the test set; while our method is pattern-based and thus we simply applied our patterns to the whole labeled corpora. The experimental results are shown in Table 3.9 and Table 3.10. Quite expectedly, PASMED is outperformed by the supervised systems, although it shows comparable performance for the LLL corpus.

Besides the parser's errors and greedy extraction presented in the previous section, the seemingly low precision scores of PASMED are caused by the system's generality. As stated before, our extraction schema covers any kinds of relations; it does not only focus on the interaction relationship. Therefore, even when the extracted relations are true, if they are not interaction relations, they are treated as false positives according to the gold-standard annotations. Figure 3.5 shows examples that PASMED extracted true relations between two proteins 'IFN-gamma' and 'IFN-alpha' in (a) and two drugs 'fluoroquinolones' and 'antibiotics' in (b), but their relationships are (a) 'associated_with' and (b) 'is_a', which are judged as false positives when compared with the annotated PPI and DDI corpora. We may improve the precision of our system by setting rules to filter out those kind of relations. For



(a)Levels of IFN-gamma were slightly increased following IFN-alpha treatment.

(b)The fluoroquinolones are a rapidly growing class of antibiotics with a broad spectrum of activity against gram-negative.

Figure 3.5: Examples of true extracted relations that are treated as false positive ones according to the annotated PPI and DDI corpora. (a) 'associated_with' relation. (b) 'is_a' relation.

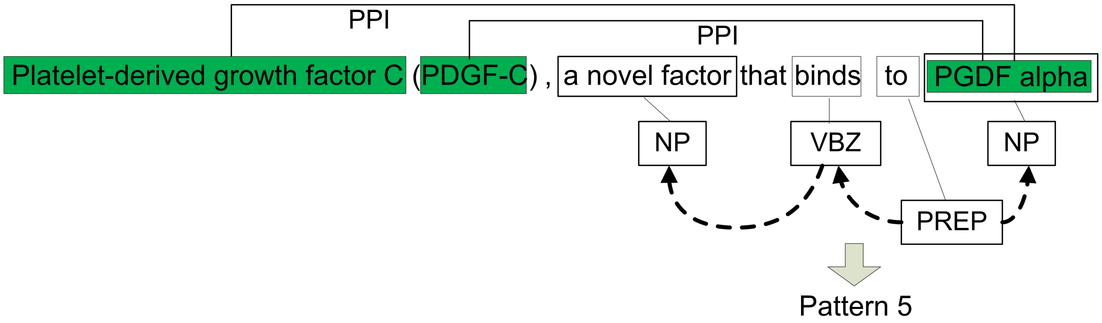

PPI
PPI
Platelet-derived growth factor C (PDGF-C) , a novel factor that binds to PGDF alpha
NP
VBZ
NP
PREP
Pattern 5

Figure 3.6: An example of two PPIs that need coreference information to be identified. Our system can detect a NP pair according to Pattern 5 but cannot extract any relations.

example, we can use a set of verbs that describe the relation of interaction, such as interact and activate, to validate the extracted relations.

The low recall scores are due to the lack of patterns and coreference resolution. Figure 3.6 illustrates an example that our system missed two PPIs since it has no information about coreference that is essential to infer them. In this example, our system can detect a NP pair of (a novel factor, PGDF alpha) according to Pattern 5. The system, then, could not identify any relation since the first NP does not contain any entity. However, in fact, there are two PPIs between 'PGDF alpha' and the two coreferences of 'a novel factor', which are 'Platelet-derived growth factor' and 'PDGF-C'.

We have investigated 100 false negative PPIs on the AIMed corpus and found that there are 21 false negative ones (21%) caused by this error. It is clear that if PASMED could perform accurate coreference resolution, it would cover more interactions. Another solution is that we can create more patterns to capture interaction expressions, such as 'an interaction between A and B', 'a complex of/between A and B', 'A-B complex', and 'A-B binding'. There are 28 false negative interactions satisfying the expressions. However, these patterns are not general enough for all type relations; they are only specific for PPI and DDI.

3.4 Large-Scale Semantic Relation Extraction

In the biomedical domain, large-scale event extraction has attracted many researchers [119, 89, 11, 120, 59, 146, 140]. Miyao et al. [89] propose a system that extracts verb-mediated relations between genes, gene products, and diseases from MEDLINE. The output of this system is served as a database for MEDIE [101], a semantic search engine on MEDLINE. Björne et al. [11] apply their system to the titles and abstracts of all PubMed citations. The extraction is performed using a pipeline composed of the BANNER named entity recognizer, the McClosky-Charniak domain-adapted parser, and the Turku Event Extraction System. Kilicoglu et al. [59] also run their system on the entire set of PubMed citations to create SemMedDB, a repository of semantic predications.

We have applied our system to the whole MEDLINE[6], which consists of about 113.6 million of sentences. We, then, counted the number of candidates and relations extracted by each PAS pattern to see its distribution in MEDLINE. Finally, we have analyzed the extracted relations and discussed about their semantic types in the context of the previous large-scale systems.

3.4.1 Distribution of PAS patterns

We have counted the matching times and the number of relations extracted from each pattern in the whole MEDLINE and report the statistical results in Table 3.11. The table shows that Pattern 6 generated the highest number of candidates and the highest number of extracted relations. However, this pattern also produced the highest number of invalid relations; it extracted about 63.28% of candidates but contributed only 36.53% of extracted relations. Compared with Pattern 6, Pattern 1 and 5 are more effective since they created significantly lower numbers of candidates but comparable numbers of relations. Furthermore, if we divide six patterns into 3 categories, including transitive verbs (Pattern 1, 2, 3, and 4), intransitive verbs (Pattern 5) and prepositions (Pattern 6), the category of transitive verbs are the most effective patterns. This group contributed only 23.04% of candidates but 41.72% of extracted relations, which outperforms the group of prepositions. We conclude that most of semantic relations in MEDLINE extracted by our system are composed by transitive verbs.

[6] http://www.nlm.nih.gov/bsd/licensee/2012_stats/baseline_med_filecount.html

Table 3.11: The distribution of our PAS patterns in MEDLINE.

Pattern	1	2	3	4	5	6
#Candidates	39.9M	6.0M	8.2M	4.8M	35.0M	162.0M
	15.58%	2.37%	3.21%	1.88%	13.68%	63.28%
#Relations	34.2M	6.6M	11.9M	4.3M	29.8M	50.0M
	24.95%	4.86%	8.73%	3.18%	21.75%	36.53%

Table 3.12: The ten most frequent types of semantic relations extracted from the whole MEDLINE.

	Semantic Relation Type		**#Rel.**	**#Uni.**
	Entity 1	Entity 2		
1	Amino Acid, Peptide or Protein	Amino Acid, Peptide or Protein	**3.4M**	1,057K
2	Cell	Amino Acid, Peptide or Protein	3.1M	711K
3	Gene or Genome	Amino Acid, Peptide or Protein	1.8M	766K
4	Disease or Syndrome	Disease or Syndrome	1.7M	599K
5	Body Part, Organ, or Organ Comp.	Amino Acid, Peptide or Protein	1.7M	561K
6	Amino Acid, Peptide or Protein	Disease or Syndrome	1.6M	631K
7	Gene or Genome	Cell	1.1M	315K
8	Organic Chemical	Organic Chemical	1.1M	365K
9	Body Part, Organ, or Organ Comp.	Body Part, Organ, or Organ Comp.	1.1M	270K
10	Laboratory Procedure	Amino Acid, Peptide or Protein	1.1M	453K

3.4.2 Analyzing Semantic Relations

Our system extracted more than 137 millions of semantic relations in the format of (entity 1, relation phrase, entity 2) from the whole MEDLINE. The ten most frequent types of relations are listed in Table 3.12. The most common semantic relation type is the relation between 'Amino Acid, Peptide or Protein' entities, which count up to 3.4 million. This explains partially why PPI has been attracting considerable attention in the BioNLP community. Many of the previous studies focus on improving PPI performance [1, 87, 67]. There are many large-scale databases constructed from MEDLINE and they focus on PPI, e.g., MedScan [29], AliBaba [102], and Chowdhary et al.[23].

Another type of relation that is also extensively studied in the community is the relation between genes and proteins, which is ranked third in Table 3.12. As with PPI, there are

Semantic Relation		**Count**
Entity 1	Entity 2	
Amino Acid, Peptide or Protein	Amino Acid, Peptide or Protein	3.4M
Gene or Genome	Amino Acid, Peptide or Protein	1.8M
Gene or Genome	Gene or Genome	793K
Nucleic Acid, Nucleoside, or Nucleotide	Amino Acid, Peptide or Protein	579K
Gene or Genome	Nucleic Acid, Nucleoside, or Nucleotide	319K
Amino Acid Sequence	Amino Acid, Peptide or Protein	172K
Nucleotide Sequence	Gene or Genome	121K
Nucleic Acid, Nucleoside, or Nucleotide	Nucleic Acid, Nucleoside, or Nucleotide	218K
Amino Acid Sequence	Gene or Genome	66K
Nucleotide Sequence	Nucleic Acid, Nucleoside, or Nucleotide	46K
Amino Acid, Peptide or Protein	Enzyme	61K
Nucleotide Sequence	Nucleotide Sequence	21K
Enzyme	Gene or Genome	19K
Nucleic Acid, Nucleoside, or Nucleotide	Enzyme	13K
Total		7.6M

Table 3.13: Statistics of protein-protein interactions in the whole MEDLINE.

many studies and databases related to this type of relations, such as Chilibot [20], MEDIE [89], EVEX [146] and the BioNLP Shared Task [98].

The second most common type of relations in our extraction result is the ones between cell and protein entities, which appeared more than 3.1 million times in MEDLINE. This type of relations contain many localization and whole-part relations, the information of which is potentially very useful in biology. These relations are covered partially by *localization events* in the GENIA corpus. The events are represented as ‘Localization of Protein to Location’ where Location can be cells. Recently, the CG task [108] has also targeted events on ‘Localization of Proteins at/from/to Cells’.

Somewhat unexpectedly, the relations between genes and diseases, which are another important type of biomedical relations [24], turned out to be much less common than PPIs. More specifically, its rank was the 41^{th} and the number of relations extracted from MEDLINE was about 583,000.

The last column in Table 3.12 shows that the diversity of the semantic relations is slightly different from their occurrences. For instance, the cell-protein relations are more frequent but less diverse than the gene-protein ones.

We grouped the following semantic types:

- Amino Acid, Peptide or Protein,
- Amino Acid Sequence,
- Enzyme,
- Gene or Genome,
- Nucleic Acid, Nucleoside, or Nucleotide,
- Nucleotide Sequence

as *protein* type. We then calculated how many types of protein-protein interactions exist in MEDLINE.

Table 3.13 describes the actual number of semantic relations of each type. In total, our system generated more than 7.6 millions of protein-protein interactions in MEDLINE. Based on our precision reported above, more than 3.5 million relations (7.6 x 0.47) are expected to be correct.

Chapter 4

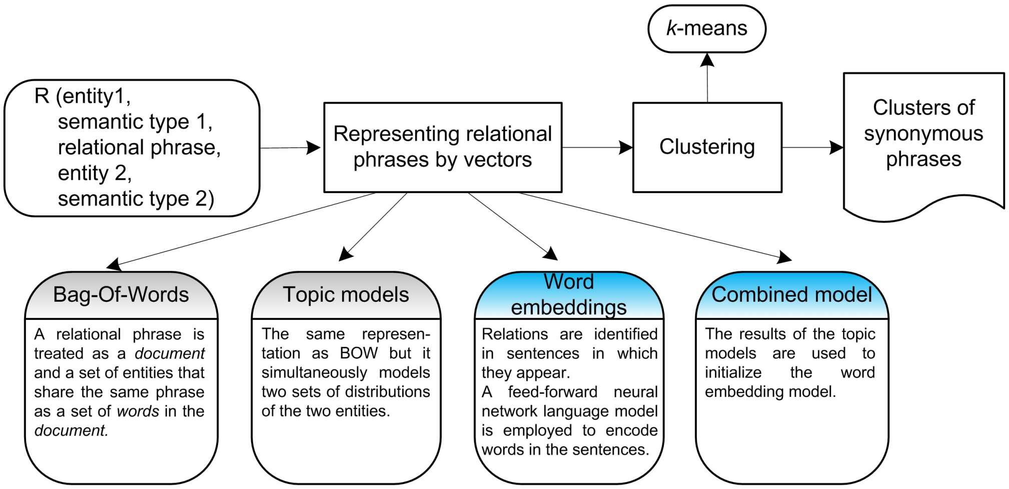
Synonym Resolution for Relational Phrases

The module of synonym resolution is input a set of relations in the format of <relational phrase, entity 1, semantic type 1, entity 2, semantic type 2> (relations that are produced by PASMED), and outputs clusters of synonymous relational phrases. To perform this task, we first encode the relational phrases into vector format by using three different unsupervised techniques: bag-of-words, topic models and word embeddings. Next, we combine the last two models by using the results of topic models to initialize the word embedding model. We then apply the k-means algorithm on top of these vector representations to cluster relational phrases into synonymous groups. An overview of our working flow is shown in Figure 4.1. We finally compare the three methods with Semantic Network Extractor (SNE) [66], a probabilistic model trained on two MLNs.

4.1 Representing Relational Phrases

4.1.1 Bag-Of-Words Model

The bag-of-words (BOW) model is a simple model commonly used in a variety of text processing tasks. In this model, a document is represented simply as a set of words regardless the syntax and even word order. Each word is represented by its index in the vocabulary and its frequency in the document. In our scenario, each relational phrase is treated as a *document* and a set of entities that share the same phrase as a set of *words* in the *document*. Consequently, a relational phrase has two bags of entities for the two corresponding arguments. The relational phrase are thus represented by a sparse vector of occurrence counts



k-means
R (entity1,
semantic type 1,
relational phrase,
entity 2,
semantic type 2)
Representing relational phrases by vectors
Clustering
Clusters of synonymous phrases
Bag-Of-Words
A relational phrase is treated as a *document* and a set of entities that share the same phrase as a set of *words* in the *document.*
Topic models
The same representation as BOW but it simultaneously models two sets of distributions of the two entities.
Word embeddings
Relations are identified in sentences in which they appear.
A feed-forward neural network language model is employed to encode words in the sentences.
Combined model
The results of the topic models are used to initialize the word embedding model.

Figure 4.1: An overview of our methods.

of entities, i.e., a sparse histogram over the vocabulary.

Assuming that our training data has one relational phrase 'treat_with' with six relation instances as presented in Table 4.1. Consequently, the vocabulary of the pseudo-training data contains eight entities as shown in Table 4.2. We index the entities from 1 to 8, respectively. As a result, the phrase 'treat_with' is represented as a vector with eight dimensions as (32, 5, 8, 19, 66, 49, 10, 7). Applying the same process for the whole training data, we obtain the vector representations for all relational phrases.

4.1.2 Topic Models

We have employed LDA-SP [122], an extension of the LinkLDA model [34], to model our relations for clustering. LDA-SP considers a relational phrase as a *document*, and a set of entities that share the same relational phrase as a set of *words* in a *document*. The advantage of this model is that it simultaneously models two sets of distributions of the entities for each topic. The graphical representation of LDA-SP is shown in Figure 4.2 (a).

In this model, each argument a_i is drawn from a different hidden topic z_i; however, the z_i's are drawn from the same distribution θ_r for a given relation r. LDA-SP allows two arguments of a given relation to be generated from $|Z|^2$ possible pairs. Since z_1 and z_2 are drawn from the same distribution θ_r, the model assigns a higher probability to states in which $z_1 = z_2$. The output of this model is the prior topic distribution of each relational phrase in R. More specifically, a relation phrase r is represented as a vector whose elements are the probabilities that the phrase belongs to a topic $p(r|t)$.

Table 4.1: A pseudo-training data that contains a relation phrase of 'treat_with' with six relation instances. The full forms of semantic types are: Disease or Syndrome (dsyn), Therapeutic or Preventive Procedure (topp), Pharmacologic Substance (phsu), and Organic Chemical (orch).

Entity 1	Sem. Type 1	Entity 2	Sem. Type 2	Freq.
Parkinson's disease	dsyn	dopaminergic drugs	phsu	5
		levodopa	phsu	8
		deep brain stimulation	topp	19
asthma	dsyn	corticosteroid	phsu	49
		montelukast	orch	10
		immunotherapy	topp	7

Table 4.2: Vocabulary of the pseudo-training data in Table 4.1.

Index	Entity	Freq.
1	Parkinson's disease	32
2	dopaminergic drugs	5
3	levodopa	8
4	deep brain stimulation	19
5	asthma	66
6	corticosteroid	49
7	montelukast	10
8	immunotherapy	7

One limitation of LDA-SP is that it only considers the surface string of the entities. For example, 'Parkinson's disease' and 'asthma' are both diseases but they are regarded as completely different entities. Exploiting the commonality of the entities belonging to the same *semantic type* [16] could therefore lead to improved performance since the types of the relations are largely determined by the semantic types of the entities.

To integrate the information on the semantic types into the LDA-SP model, we made a small modification to the model. In what follows, we call the resulting model LDA-SP-sem. LDA-SP-sem defines two roles for the semantic types (st_1, st_2) of the entities as illustrated in Figure 4.2 (b). It should be noted that the argument and its semantic type have different

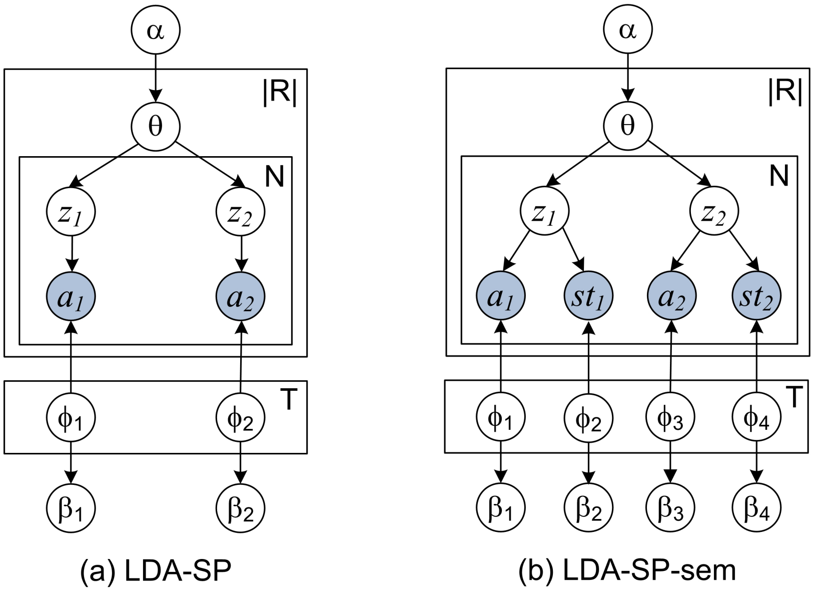


Figure 4.2: The graphical representations of (a) the LDA-SP [122] model and (b) the LDA-SP-sem model. The LDA-SP-sem model sets different roles for the semantic type of each argument. The semantic types are constrained to be in the same topic as their entities.

word distributions but they are drawn from the same hidden topic. For instance, referring to the pseudo-training data in Table 4.1, the word distribution of a_1 is (Parkinson's disease: 32, asthma: 66), while that of st_1 is (dsyn: 98).

We implemented the LDA models by using collapsed Gibbs sampling [45] for inference[1].

4.1.3 Word Embeddings

Mikolov et al. [80] introduced two effective techniques for learning vector representations of words from large amounts of unstructured text data: the Continuous Bag-Of-Word (CBOW) model and the continuous Skip-gram model.

The CBOW model is similar to the feed-forward neural network language model [9], where there is no hidden layer and the projection layer is shared for all words. Unlike the BOW model, this model predicts a word by using the continuous context around that word. Given a sequence of training words $w_1, w_2, w_3 ... w_T$, the objective of this model [82] is to maximize the average log probability as shown in Equation 4.1, where c is the size of the

[1] The LDA models are implemented by Makoto Miwa.

context window.

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_t|w_{t+j}) \tag{4.1}$$

The basic model defines $p(w_t|w_{t+j})$ using a softmax function:

$$p(w_O|w_I) = \frac{\exp({v'_{w_O}}^{\top} v_{w_I})}{\sum_{w=1}^{V}\exp({v'_w}^{\top} v_{w_I})} \tag{4.2}$$

where v_w and v'_w are the *input* and *output* vector representation of the word w, and V is the number of words in the vocabulary. In contrast with the CBOW model, the Skip-gram model receives the current word and predicts words within a certain window.

In this work, we use the CBOW model[2] to estimate vector representations of our relational phrases. More specifically, for each relation in the format of <entity 1, relational phrase, entity 2>, which is given by PASMED, we retrieve the sentence that contains the relation from the original text database. We then identify the words or phrases that correspond to the entities and relational phrases, and create newly-defined *words* for them depending on their roles in the relation. For example, the relation of <parkinson's disease, treat with, dopaminergic drug> in the sentence "Many patients with Parkinson's disease are treated with dopaminergic drugs" will produce the following training example for the CBOW model:

"many patient with parkinson's_disease@arg1 be treat_with@pred dopaminergic_drug@arg2"

Note that multi-word terms are grouped with underscores and the roles in the relation are indicated by artificial suffixes such as '@arg1'.

After training the CBOW model, we extract the distributed feature vector v'_w associated with each relational phrase w and apply k-means clustering to them.

4.2 Evaluation Settings

4.2.1 Data

Training data More than 70 millions of biomedical relations in the format of <relational phrase, entity 1, semantic type 1, entity 2, semantic type 2> were extracted from MEDLINE in a period of 2004-2012 by PASMED (described in Chapter 3). We normalized our data by removing relations whose

- relational phrase is not a verb or a verb phrase,

[2] The model is implemented in the word2vec tool: https://code.google.com/p/word2vec/

- entity 1 or entity 2 is not composed by continuous words, or
- occurrence is lower than 5.

As a result, our data consists of 763,065 unique relations and 7,132 unique relational phrases. All entities and relational phrases were stemmed and lower-cased before training.

Evaluation data To evaluate our clustering results, we created a gold standard of synonymous groups based on Nebot and Berlanga's data [97]. We stemmed every phrase, discarded duplicate terms in each group, and removed groups that have only one term. As a result, our gold standard consists of 286 relational phrases clustered into 100 non-singleton groups with an average cluster size of 3.9.

4.2.2 Perplexity

There are several metrics that can be used for evaluating topic models [148]. In this work, we use the perplexity on the training and testing set. Formally, for a set S of M documents, perplexity is calculated as Equation 4.3 [15], in which $p(w_m)$ is computed according to the value θ of the model.

$$Per(S) = \exp\left(-\frac{\sum_m \log p(w_m)}{\sum_m |w_m|}\right) \tag{4.3}$$

The lower the perplexity, the better the model.

4.2.3 Resolver Metric

The precision, recall and F scores of our generated clusters were computed by using the Resolver metric [159]. The scores are calculated by measuring the overlap of the best matches between the gold standard groups and the non-singleton generated clusters, i.e., clusters that have more than one term. Precision is the fraction of terms in the generated clusters which are also in the gold standard. It should be noted that the terms that are not included in the gold standard were removed from the data set before evaluating precision. Similarly, recall is computed by swapping the roles of the generated and the gold standard clusters. We finally used micro-averaging to calculate the global scores.

For example, we have three gold standard clusters of (a b), (c d e f g), (h k l) and two generated clusters of (a b c d f g), (e h k l). When we map from the generated clusters to the gold standard, the best matches are (c d f g) and (h k l). Hence, the precision is 7/10. Vice versa, the best matches from the gold standard to the generated clusters are (a b), (c d f g) and (h k l), which produce a recall of 9/10.

Table 4.3: Perplexity of LDA-SP and LDA-SP-sem on the training and testing sets.

Model	Set	Number of topics				
		10	50	100	200	300
LDA-SP	Training	722.6	476.2	414.5	370.0	347.5
	Testing	688.8	452.2	394.0	352.9	331.2
LDA-SP-sem	Training	690.2	448.7	390.6	350.0	330.9
	Testing	688.3	448.7	392.3	352.7	333.3

4.2.4 CBOW Configurations

For a fair comparison, we set the dimension of the CBOW model based on the number of topics used in the topic models. We select a hierarchical softmax classification for the output layer and a context window size of 5.

4.3 Experimental Results

We first conduct experiments to find out the suitable number of topics for the LDA models. Then a sequence of experiments on clustering is carried out to evaluate the performance of our approaches and compare them with that of SNE.

4.3.1 Perplexity of The LDA Models

In this evaluation, we divided our training data into 10 parts; 9 parts were used for training and the other part for testing. Elements in the training and testing sets share the same indices of relational phrases.

We ran 2,000 iterations for inference with a varied number of topics and obtained the corresponding perplexity results of training and testing sets in Table 4.3. These results show that the perplexities of LDA-SP-sem are slightly lower than those of the LDA-SP on the training set but they are comparable each other on the testing set. LDA-SP-sem exhibits overfitting in contrast to LDA-SP since its perplexities on the testing set are higher than those on the training set.

In both models, the perplexities decreased when the number of topics increased but they did not substantially change from 200 topics. Hence, we used the output of 200 topics for our clustering step and the dimension of the CBOW model was set to 200.

Table 4.4: Clustering results when BOW, LDA-SP and LDA-SP-sem are used to represent relational phrases.

k	BOW			LDA-SP			LDA-SP-sem		
	Pre.	Re.	F.	Pre.	Re.	F.	Pre.	Re.	F.
100	18.94	23.13	20.77	29.73	20.88	24.52	22.38	17.79	19.75
200	20.80	21.65	21.17	35.21	18.62	24.36	26.18	14.60	18.74
300	23.19	19.88	21.38	36.39	17.30	23.44	26.88	12.30	17.52
400	24.36	18.95	21.27	37.91	16.95	23.41	27.86	12.20	16.97
500	26.27	17.77	21.15	39.54	16.41	23.19	29.14	11.46	16.44

4.3.2 Clustering Results

After representing the relational phrases as vectors, we applied k-means clustering in Bayon[3] on top of those vectors with varying numbers of clusters (k). For each value of k, we run k-means with 10 random seeds and calculate the mean scores. We also compare our methods with Semantic Network Extractor (SNE) [66], a probabilistic model based on two MLNs.

Experimental results in Table 4.4 indicate that BOW boosts the recall while LDA-SP and LDA-SP-sem boost the precision. Our error analysis shows that BOW usually produces clusters that can cover different gold clusters. For example, the two gold clusters, (activate, initiate, stimulate, trigger) and (affect, induce, inhibit, suppress) are grouped into one cluster by BOW with k=100. This grouping leads to a higher recall but might affect the precision.

In terms of F-scores, LDA-SP is slightly better than BOW, but LDA-SP-sem, unexpectedly, yields the worst performance. Based on the above example, the low recall of LDA-SP-sem can be explained by the fact that the model separates gold clusters into many clusters. For instance, (affect, induce, inhibit, suppress) is distributed into two different groups; one group contains (induce, inhibit, suppress) and the other contains 'affect'. Among the three models, the highest performance is an F-score of 24.52%, produced by LDA-SP when k is 100.

In case of SNE, we directly input more than 763 thousand unique relations to produce clusters of synonymous strings. SNE[4] allows us to tune three parameters: the total value of $\alpha + \beta$, λ, and μ. We started with the empirical values reported in [66], which are 10, 100, and 100 respectively. According to Equation 2.3 (Chapter 2), the number of non-singleton

[3] https://code.google.com/p/bayon/

[4] http://alchemy.cs.washington.edu/papers/kok08/

Table 4.5: Clustering results of SNE with varying values of ($\alpha + \beta$, λ, μ).

Values of parameters	Pre.	Re.	F.
(10, 100, 100)	21.01	24.81	22.75
(20, 200, 200)	23.68	25.06	24.35
(30, 300, 300)	22.81	23.53	23.16
(40, 400, 400)	19.33	21.23	20.23
(50, 500, 500)	19.67	23.02	21.21

clusters will be increased if we increase the value of the three parameters. Hence, we tuned those values in increments of 10, 100 and 100 to find out the best performance. Table 4.5 shows that SNE produced lower precision but slightly higher recall than LDA-SP on our data set. The best score of SNE is 24.35%, where the three parameters are 20, 200, and 200 respectively.

Regarding word embeddings, we investigate the performance of the CBOW model with three different representations of a relation:

(i) *Relation*: treating a relation as a sentence. This representation uses the same information as BOW, LDA-SP, and SNE.

(ii) *Sentence*: embedding the relation in the sentence in which it appears and assigning a role to the relational phrase.

(iii) *Role*: embedding the relation in the sentence in which it appears and assigning corresponding roles to the relational phrase and its two entities.

For instance, a relation of <parkinson's disease, treat with, dopaminergic drug> extracted from the sentence "Many patients with Parkinson's disease are treated with dopaminergic drugs" will be represented by three ways shown in Table 4.6.

Table 4.7 presents the size of vocabulary and the number of words in the training data corresponding to each representation. It is reasonable that by assigning roles to entities we increased the size of vocabulary and the number of words in the training phase, i.e., the training data is sparser. In case of the *Relation* representation, since we do not use the context around a relation, the vocabulary and words are substantially lower than the others.

The experimental results of clustering are shown in Table 4.8. The performance of each representation is not consistent in terms of the value of k. The highest scores were obtained

Table 4.6: Three ways of modeling a relation of <parkinson's disease, treat with, dopaminergic drug>.

Type	Representation
Relation	"parkinson's_disease treat_with dopaminergic_drug"
Sentence	"many patient with parkinson's_disease be treat_with@pred dopaminergic_drug"
Role	"many patient with parkinson's_disease@arg1 be treat_with@pred dopaminergic_drug@arg2"

Table 4.7: Vocabulary size and number of words by each representation.

	Relation	*Sentence*	*Role*
Vocabulary size	340K	494K	653K
Number of words	126M	268M	268M

by 100 clusters for *Role*, 200 clusters for *Sentence*, and 300 clusters for *Relation*. Among them, the *Role* representation performs slightly better than the others despite the fact that this representation make the training data sparser.

Our observation shows that this type of representation generated more correct clusters. For example, with *Sentence* and *Relation*, three strings 'infect', 'be infectious for' and 'infest' were assigned to two different clusters. However, in case of *Role*, those strings were grouped in one clusters, which is more accurate according to the gold standard. Among the three representations, *Sentence* and *Role* can capture the continuous context around relations while *Relation* cannot. Therefore, these two representations yield better results than *Relation*.

Compared with BOW, SNE, and LDA-SP, CBOW boosts the performance of clustering on both precision and recall scores. CBOW tends to produce more correct synonymous terms in clusters. For instance, it can assign eleven verbs of laboratory procedures into one group, while the other methods can partially do it, i.e., they can assign at most six terms into one group, as illustrated in Table 4.9. It is clear that by using word embeddings, the performance of clustering was improved significantly.

We collect the highest performance figures of each method and show them in Table 4.10. χ^2 tests with one degree of freedom were conducted on the precision and recall of three pairs of methods including SNE vs. CBOW-*Relation*, LDA-SP vs. CBOW-*Relation*, and

Table 4.8: Clustering results when the CBOW model is used to learn relational phrases' vectors.

k	*Relation*			*Sentence*			*Role*		
	Pre.	Re.	F.	Pre.	Re.	F.	Pre.	Re.	F.
100	32.48	32.32	32.38	29.56	35.20	32.09	29.19	36.50	32.33
200	34.39	26.69	30.01	36.17	31.60	33.66	34.18	32.69	33.35
300	36.92	25.99	30.50	38.85	28.62	32.93	39.80	29.95	34.14
400	37.42	25.11	30.02	40.03	27.95	32.89	39.49	28.09	32.80
500	41.25	24.41	30.65	41.55	26.69	32.44	42.54	28.29	33.95

Table 4.9: An example of clustering verbs that convey laboratory procedures by the four methods. The italic phrases are incorrect terms according to the gold standard.

Method	Clustering result
BOW	analyse, assess, examine, evaluate, estimate, test
LDA-SP	analyze, assess, examine, evaluate, investigate, test
SNE	assess, examine, evaluate, measure, *compare, confirm, detect*
CBOW	analyse, analyze, assay, assess, define, estimate, evaluate, examine, investigate, measure, test, *characterise, characterize, compare, determine, map*

CBOW-*Relation* vs. CBOW-*Role*. Regarding the first pair, we gained p-value < 0.05 for both precision and recall. With LDA-SP vs. CBOW-*Relation*, the p-value was less than 0.05 in case of recall, while this happened in case of precision for CBOW-*Relation* vs. CBOW-*Role*. These results can be interpreted as (1) when using the same information as SNE and LDA-SP, the CBOW model performs significantly better than the two methods; and (2) the precision is further improved by embedding the relations into sentences with keeping their roles.

4.3.3 Combining Word Embeddings and LDA-SP

As shown in Equation 4.2, the CBOW model first initializes the input vector representations v_w of the word w and then learns the output vector v'_w based on the training data. Instead of initializing the input vectors randomly for the CBOW model, we use the output vectors

Table 4.10: The highest performance of each method on our evaluating data.

Method	Feature	Pre.	Re.	F
BOW	Relations	23.19	19.88	21.38
SNE	Unique Relations	23.68	25.06	24.35
LDA-SP	Relations	29.73	20.88	24.52
CBOW-*Relation*	Relations	32.48	32.32	32.38
CBOW-*Sentence*	Embedded relations	36.17	31.60	33.66
CBOW-*Role*	Embedded relations with roles	39.80	29.95	34.14

Table 4.11: Clustering results when using LDA-SP's output vectors to initialize CBOW.

k	*Relation*			*Sentence*			*Role*		
	Pre.	Re.	F.	Pre.	Re.	F.	Pre.	Re.	F.
100	31.98	31.74	31.80	28.85	35.64	31.80	29.24	35.87	32.15
200	37.53	28.16	32.14	33.25	29.88	31.43	33.82	30.22	31.89
300	38.14	26.18	31.02	37.09	27.95	31.85	37.37	29.27	32.81
400	40.37	24.92	30.78	38.94	27.06	31.89	39.17	26.95	31.91
500	40.25	23.44	29.60	40.61	25.53	31.33	40.50	27.06	32.42

of the LDA-SP. The experimental results in Table 4.11 show that although we used a smart initialization, the performance of clustering could not be improved further. Compared with the CBOW model, the combined model failed to identify some synonymous phrases. For instance, the CBOW model can assign (be sensitive to, sensitise, sensitize) in one cluster, while the combined model can detect only a pair of 'sensitise' and 'sensitize'. These preliminary results indicate that the output vectors by LDA-SP might not be suitable for initializing the CBOW model and finding a solution for this will be follow-up work.

4.4 Discussion

Unlike previous work that tried to cluster both entities and relational phrases [66, 159], our work only aims at clustering the phrases. However, since we treated entities in relations as *words* in *sentences*, the trained model by CBOW can also be used to cluster entities. By calculating the cosine similarity between vector representations of entities, we can detect similar or synonymous entities. For instance, according to our model, the closet to the entity ‘gastric_cancer’ are ‘gastric_carcinoma’ and ‘gastric_adenocarcinoma’ with a value of 0.82, and indeed they are synonymous. This is an advantage that does not exist in BOW or LDA-SP.

Moreover, we have found that the property of algebraic operations on vector representations is maintained in this task. As stated above, we group continuous relational phrases as *words*, but not with discontinuous phrases. For example, the relational phrase between entities in the the following sentence is discontinuous.

“... we **investigate** surviving messenger RNA MRNA expression **in** gastric_cancer ...” However, as expected, $vector$(investigate) + $vector$(in) is close to vectors of ‘investigate_in’, ‘assess_in’, and ‘evaluate_in’, which means that they are similar phrases. This property, again, confirms the robustness of the CBOW model in comparison with BOW, SNE and LDA-SP.

The highest empirical F-score achieved in our experiments was 34.14%. This is not an ideal level of performance but at the same time is an encouraging performance figure, considering that the clustering is done in a fully unsupervised fashion and the evaluation criteria are strict. An interesting line of future work would be to incorporate some level of supervision to further improve the clustering accuracy. In Table 10, we show some clusters of relational phrases obtained by our model, in which most of the phrases are indeed synonymous.

These synonymous clusters will be useful for question-answering systems that support natural language queries such as Linked Open Data Question-Answering (LODQA)[5]. Assuming that the system queries on a database of general relations output by our Open IE system (PASMED). When we input a query of “What genes are essential for cell survival?”, this system first generates a predicate-argument relation graph and creates a pseudo SPARQL query as follows:

```
SELECT ?t1
WHERE {
   ?t1 [:isa] [genes] .
   ?t2 [:isa] [cell survival] .
   ?t1 [be essential for] ?t2 .
}
```

As a result, the system will return 58 unique relations in which the semantic type of the first entity is gene, the second entity is ‘cell survival’, and the relational phrase is ‘be essential

[5] Currently, LOQDA (http://lodqa.dbcls.jphttp://lodqa.dbcls.jp) queries on the Online Mendelian Inheritance in Man (OMIM) database.

Table 4.12: Examples of good clusters of relational phrases. Each cluster is assigned a name that conveys its meaning.

Laboratory procedures	analyse at, analyze at, ascertain at, assess at, collect at, compare at, determine at, do at, evaluate at, examine at, exercise at, harvest at, identify at, investigate at, isolate at, measure at, monitor at, note at, obtain at, perform at, record at, remove at, sample at, screen at, study at, take at, test at
Localization relations	accumulate at, be localized in, be localized to, bud at, cluster at, colocalise with, colocaliz in, colocaliz with, colocalize in, co-localize in, co-localize to, colocalize with, co-localize with, colocalize within, concentrate at, concentrate in, enrich at, enrich on, localise in, localise to, localize at, localize in, localize on, localize to, localize with, localize within, localized to, locate to, recruit to, shuttle between, target to, translocate from, translocate into, translocate to
Necessity relations	be central in, be central to, be critical for, be critical in, be critical to, be crucial for, be crucial in, be crucial to, be dispensable for, be essential for, be essential in, be essential to, be fundamental to, be important for, be important in, be important to, be instrumental in, be integral to, be key to, be necessary for, be pivotal in, be sufficient for, contribute to, cooperate in, function in, involve in, participate in, require for

for'. However, if we use the synonymous cluster of necessity relations (the third row in Table 4.12), the search term can be expanded and the number of the answers would be increased to 261. We, therefore, conclude that the synonymous clusters would help the QA system to find more results.

Another application-level example is applying synonymous groups to entailment detection. Rei and Briscoe [112] defined four entailment relations between two fragments A and B: A → B, B → A, A = B, and A ≠ B. Our synonymous groups can be directly used for the third relation and for expanding results of the other relations. For instance, according to their pilot dataset[6], there is an entailment relation as "investigates = examines", which is identical to our synonymous pair (investigate, examine). Also, an entailment relation

[6] http://www.marekrei.com/?cat=projects&page=fragmentail

between "stimulate → affect" can be expanded to "activate → affect" since we know that 'activate' is a synonym of 'stimulate'.

One of the limitations of our work is that we only focus on hard clustering, i.e., a phrase is assumed to be in only one cluster. However, in practice, a phrase can belong to more than one cluster when it is polysemous. For instance, there is about 26% of polysemous phrases, which occupy about 47% of occurrences, in the evaluating data. The output vector of the LDA-SP model can be interpreted as a result of soft clustering, in which LDA-SP assigns, for instance, a probability of 0.3 for topic 1, 0.15 for topic 2, 0.4 for topic 3 ..., to a phrase a. Let consider the topics as senses of a phrase. If we set a threshold of 0.2, the phrase a will belong to senses 1 and 3. But, if we set a threshold of 0.5, the phrase a has no sense. Ideally, for a polysemous phrase, instead of assigning a probability to each sense, the method should assign the probability of having more than two senses. This issue may be addressed by using statistical models for partial membership [50], but we leave it for future work.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this dissertation, we have developed PASMED to extract diverse types of relations between biomedical entities from the literature. Six simple but effective PAS patterns have been proposed to detect relevant NP pairs. Our evaluation results have confirmed that our pattern-based system covers a wide range of relations. Although the precision scores of PASMED fell short of those of SemRep, the overall results suggest that PASMED compares favorably with SemRep, extracting a significantly higher number of relations. We have applied PASMED to the entire MEDLINE corpus and extracted 137 million semantic relations. This large-scale and machine-readable output can be used to scale-up high-quality manual curation of a relation ontology or served as a knowledge base for semantic search.

Our extraction schema is limited in several ways. First, the filtering process discards frequent named entities, which causes the missing of relations that involving those entities. Second, there is no coreference resolution module incorporated into the system, therefore the system cannot identify relations that are inferred based on coreference information. And third, since the PAS patterns only focus on verbs and prepositions, they cannot cover other complex predicate types, e.g, nominalizations.

After extracting general relations from MEDLINE, we next perform synonymy detection for relational phrases that represent the relations. Four unsupervised methods were applied to cluster relational phrases. The first three methods, BOW, LDA-SP and CBOW, encode relational phrases into vector format, while SNE approaches the task by using a probabilistic model enhanced with two Markov logic networks. Our experimental results on a part of the relations extracted from MEDLINE indicate that CBOW significantly outperforms BOW, LDA-SP and SNE. This finding confirms the effectiveness of using word embeddings to detect synonymous phrases. We also tried initialize CBOW by using the output from LDA-SP but the combined model unexpectedly performed worse than the CBOW model alone. Our observation on the best clustering result has revealed some synonymous groups that will be useful for high-level tasks in biomedical text mining, e.g., question answering and entailment detection.

A limitation of our work is that we currently ignore soft clustering, i.e., we assume that each relational phrase belongs to a cluster and do not concern about polysemous phrases. Addressing soft clustering may be a follow-up work in the near future.

5.2 Future Work

We wish to extend our system to address *n*-ary relations [2, 75]. Relations of this type are more informative than binary ones since they can include details about the site, context or conditions under which biomedical relations occur. For example, given the following sentence: “The integrated [stress response] is activated by [halofuginone] in [mammary epithelial cells]”, our system can detect two binary relations of (stress response, halofuginone) and (halofuginone, mammary epithelial cells). However, in this case, it should be better if we can identify an *n*-ary relation composed by the three entities, which describes that the activation of ‘stress response’ and ‘halofuginone’ happens in a specific location of ‘mammary epithelial cells’, not in any location. By presenting more details about relations, the n-ary relations can be treated as biomedical events.

Another future work would be to build a real text-mining application that allows end-users to browse or retrieve information about biomedical relations using queries in natural language like LODQA. Basically, the application can employ the Enju parser to analyze the queries, and then match the parsed components with the whole collection of relations extracted from MEDLINE. The application can also make use of the synonymous relational phrases to extend its search terms. Additionally, we need to create an interface between the users and the application, and to implement a good string matching algorithm. This application would help biologists and researchers in the biomedical domain to search information quickly and accurately.

Bibliography

[1] A. Airola, S. Pyysalo, J. Björne, T. Pahikkala, F. Ginter, and T. Salakoski. A graph kernel for protein-protein interaction extraction. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*, BioNLP '08, pages 1–9, 2008.

[2] A. Akbik and A. Löser. KrakeN: N-ary Facts in Open Information Extraction. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*, pages 52–56, 2012.

[3] B. Alex, B. Haddow, and C. Grover. Recognising Nested Named Entities in Biomedical Text. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*, pages 65–72, 2007.

[4] R. Altman, C. Bergman, J. Blake, C. Blaschke, A. Cohen, F. Gannon, L. Grivell, U. Hahn, W. Hersh, L. Hirschman, L. Jensen, M. Krallinger, B. Mons, S. O'Donoghue, M. Peitsch, D. Rebholz-Schuhmann, H. Shatkay, and A. Valencia. Text mining for biology - the way forward: opinions from leading scientists. *Genome Biology*, 9(Suppl 2):S7, 2008.

[5] R. K. Ando. BioCreative II Gene Mention Tagging System at IBM Watson. In *Proceedings of The Second BioCreative Challenge Evaluation*, pages 101–103, 2007.

[6] A. R. Aronson and F.-M. Lang. An overview of MetaMap: historical perspective and recent advances. *JAMIA*, 17(3):229–236, 2010.

[7] C. F. Baker, C. J. Fillmore, and J. B. Lowe. The Berkeley FrameNet project. In *COLING-ACL '98*, pages 86–90, Montreal, Canada, 1998.

[8] M. Banko, M. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni. Open information extraction from the web. In *Proceedings of IJCAI*, pages 2670–2676, 2007.

[9] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin. A Neural Probabilistic Language Model. *Journal of Machine Learning Research*, 3:1137–1155, 2003.

[10] D. M. Bikel, S. Miller, R. Schwartz, and R. Weischedel. Nymble: a High-Performance Learning Name-finder. In *In Proceedings of the Fifth Conference on Applied Natural Language Processing*, pages 194–201, 1997.

[11] J. Björne, F. Ginter, S. Pyysalo, J. Tsujii, and T. Salakoski. Scaling up Biomedical Event Extraction to the Entire Pubmed. In *Proceedings of BioNLP'10*, pages 28–36, 2010.

[12] J. Björne, J. Heimonen, F. Ginter, A. Airola, T. Pahikkala, and T. Salakoski. Extracting Complex Biological Events with Rich Graph-Based Feature Sets. In *Proceedings of the BioNLP 2009 Workshop Companion Volume for Shared Task*, pages 10–18. Association for Computational Linguistics, June 2009.

[13] J. Björne and T. Salakoski. Generalizing Biomedical Event Extraction. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, BioNLP Shared Task '11, pages 183–191. Association for Computational Linguistics, 2011.

[14] J. Björne and T. Salakoski. TEES 2.1: Automated Annotation Scheme Learning in the BioNLP 2013 Shared Task. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 16–25, Sofia, Bulgaria, August 2013. Association for Computational Linguistics.

[15] D. M. Blei, A. Y. Ng, M. I. Jordan, and J. Lafferty. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[16] O. Bodenreider. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32(suppl 1):D267–D270, 2004.

[17] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250, 2008.

[18] R. C. Bunescu and R. J. Mooney. A Shortest Path Dependency Kernel for Relation Extraction. In *Proceedings of the Human Language Technology Conference*, 2005.

[19] H. Chen, S. R. K. Branavan, R. Barzilay, and D. R. Karger. Global Models of Document Structure Using Latent Permutations. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, pages 371–379, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[20] H. Chen and B. M. Sharp. Content-rich biological network constructed by mining PubMed abstracts. *BMC Bioinformatics*, 5:147, 2004.

[21] N. A. Chinchor. Overview of MUC-7/MET-2. In *Proceedings of the* 7^{th} *Message Understanding Conference (MUC7)*, 1998.

[22] W. C. Chou, R. T. H. Tsai, and Y. S. Su. A semi-automatic method for annotating a biomedical proposition bank. In *Proceedings of FLAC'06. ACL*, 2006.

[23] R. Chowdhary, J. Zhang, and J. S. Liu. Bayesian inference of protein-protein interactions from biological literature. *Bioinformatics*, 25(12):1536–1542, 2009.

[24] H.-W. Chun, Y. Tsuruoka, J.-D. Kim, R. Shiba, N. Nagata, T. Hishiki, and J. Tsujii. Extraction of Gene-Disease Relations from Medline Using Domain Dictionaries and Machine Learning. In *Proceedings of Pacific Symposium on Biocomputing*, pages 4–15. World Scientific, 2006.

[25] S. J. Clarke and P. Willett. Estimating the recall performance of web search engines. *Aslib Proceedings*, 49(7):184–189, 1997.

[26] A. M. Cohen and W. R. Hersh. A survey of current work in biomedical text mining. *Briefings in Bioinformatics*, 6(1):57–71, March 2005.

[27] K. B. Cohen and L. Hunter. A critical review of PASBio's argument structures for biomedical verbs. *BMC Bioinformatics*, 7(Suppl 3):S5, 2006.

[28] N. Collier, S. Doan, A. Kawazoe, R. M. Goodwin, M. Conway, Y. Tateno, Q.-H. Ngo, D. Dien, A. Kawtrakul, K. Takeuchi, M. Shigematsu, and K. Taniguchi. BioCaster: detecting public health rumors with a web-based text mining system. *Bioinformatics*, 24(24):2940–2941, 2008.

[29] N. Daraselia, A. Yuryev, S. Egorov, S. Novichkova, A. Nikitin, and I. Mazo. Extracting human protein interactions from MEDLINE using a full-sentence parser. *Bioinformatics*, 20(5):604–611, 2004.

[30] H. Daumé, III and D. Marcu. Bayesian Query-focused Summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, pages 305–312, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[31] D. Davidov and A. Rappoport. Unsupervised Discovery of Generic Relationships Using Pattern Clusters and its Evaluation by Automatically Generated SAT Analogy Questions. In *Proceedings of the 46th Annual Meeting of the ACL and HLT (ACL-HLT-08)*, pages 692–700, 2008.

[32] J. Ding, D. Berleant, D. Nettleton, and E. S. Wurtele. Mining MEDLINE: Abstracts, sentences, or phrases? In *Pacific Symposium on Biocomputing*, pages 326–337, 2002.

[33] G. Doddington, A. Mitchell, M. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. The Automatic Content Extraction (ACE) Program–Tasks, Data, and Evaluation. *Proceedings of LREC 2004*, pages 837–840, 2004.

[34] E. Erosheva, S. Fienberg, and J. Lafferty. Mixed-membership models of scientific publications. *Proceedings of the National Academy of Sciences*, 101(suppl. 1):5220–5227, 2004.

[35] A. Fader, S. Soderland, and O. Etzioni. Identifying Relations for Open Information Extraction. In *Proceedings of EMNLP*, pages 1535–1545, 2011.

[36] J. R. Finkel, T. Grenager, and C. D. Manning. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pages 363–370, 2005.

[37] J. R. Finkel and C. D. Manning. Nested Named Entity Recognition. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 141–150, 2009.

[38] J. Fleiss. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382, 1971.

[39] C. Friedman, P. O. Alderson, J. H. Austin, J. J. Cimino, and S. B. Johnson. A general natural-language text processor for clinical radiology. *Journal of the American Medical Informatics Association : JAMIA*, 1(2):161–174, 1994.

[40] C. Friedman, P. Kra, H. Yu, M. Krauthammer, and A. Rzhetsky. GENIES: a natural-language processing system for the extraction of molecular pathways from journal articles. In *ISMB (Supplement of Bioinformatics)*, pages 74–82, 2001.

[41] K. Fukuda, T. Tsunoda, A. Tamura, and T. Takagi. Toward Information Extraction: Identifying protein names from biological papers. In *Proceedings of the Pacific Symposium on Biocomputing*, 1998.

[42] K. Fundel, R. Küffner, and R. Zimmer. RelEx - Relation extraction using dependency parse trees. *Bioinformatics*, 23(3):365–371, 2007.

[43] D. Gildea and M. Palmer. The necessity of parsing for predicate argument recognition. In *Proceedings of ACL*, pages 239–246, Philadelphia, Pennsylvania, USA, July 2002.

[44] A. M. Green. Kappa statistics for multiple raters using categorical classifications. In *Proceedings of the Twenty-Second Annual Conference of SAS Users Group*, 1997.

[45] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl. 1):5228–5235, April 2004.

[46] Z. GuoDong, S. Jian, Z. Jie, and Z. Min. Exploring Various Knowledge in Relation Extraction. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 427–434, 2005.

[47] A. Haghighi and D. Klein. Coreference Resolution in a Modular, Entity-Centered Model. In *HLT-NAACL*, pages 385–393. The Association for Computational Linguistics, 2010.

[48] T. Hasegawa, S. Sekine, and R. Grishman. Discovering Relations among Named Entities from Large Corpora. In *Proceedings of the Annual Meeting of Association of Computational Linguistics (ACL 04)*, 2004.

[49] K. Hashimoto, M. Miwa, Y. Tsuruoka, and T. Chikayama. Simple Customization of Recursive Neural Networks for Semantic Relation Classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1372–1376, 2013.

[50] K. A. Heller, S. Williamson, and Z. Ghahramani. Statistical Models for Partial Membership. In *Proceedings of the 25th International Conference on Machine learning*, pages 392–399, 2008.

[51] L. Hirschman, A. S. Yeh, C. Blaschke, and A. Valencia. Overview of BioCreAtIvE: critical assessment of information extraction for biology. *BMC Bioinformatics*, 6(S-1), 2005.

[52] R. Hoffmann, C. Zhang, X. Ling, L. S. Zettlemoyer, and D. S. Weld. Knowledge-Based Weak Supervision for information Extraction of Overlapping Relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 541–550, 2011.

[53] R. Hoffmann, C. Zhang, and D. S. Weld. Learning 5000 Relational Extractors. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 286–295, 2010.

[54] N. Indurkhya and F. J. Damerau, editors. *Handbook of Natural Language Processing.* Chapman & Hall/CRC, 2 edition, 2010.

[55] M. Iyyer, J. Boyd-Graber, L. Claudino, R. Socher, and H. Daumé III. A Neural Network for Factoid Question Answering over Paragraphs. In *Empirical Methods in Natural Language Processing*, 2014.

[56] D. Jurafsky and J. H. Martin. *Speech and Language Processing (2nd Edition) (Prentice Hall Series in Artificial Intelligence).* Prentice Hall, 2 edition, 2008.

[57] J. Kazama, T. Makino, Y. Ohta, and J. ichi Tsujii. Tuning support vector machines for biomedical named entity recognition. In *ACL Workshop on Natural Language Processing in the Biomedical Domain*, pages 1–8. ACL, 2002.

[58] B. Kemper, T. Matsuzaki, Y. Matsuoka, Y. Tsuruoka, H. Kitano, S. Ananiadou, and J. Tsujii. PathText: a text mining integrator for biological pathway visualizations. *Bioinformatics [ISMB]*, 26(12):374–381, 2010.

[59] H. Kilicoglu, D. Shin, M. Fiszman, G. Rosemblat, and T. C. Rindflesch. SemMedDB: a PubMed-scale repository of biomedical semantic predications. *Bioinformatics*, 28(23):3158–3160, 2012.

[60] J.-D. Kim, T. Ohta, S. Pyysalo, Y. Kano, and J. 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*, BioNLP '09, pages 1–9, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[61] J.-D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii. GENIA corpus - a semantically annotated corpus for bio-textmining. In *ISMB (Supplement of Bioinformatics)*, pages 180–182, 2003.

[62] J.-D. Kim, T. Ohta, Y. Tsuruoka, Y. Tateisi, and N. Collier. Introduction to the Bio-entity Recognition Task at JNLPBA. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and Its Applications*, JNLPBA '04, pages 70–75. Association for Computational Linguistics, 2004.

[63] P. Kingsbury, M. Palmer, and M. Marcus. Adding Semantic Annotation to the Penn Treebank. In *Proceedings of HLT*, 2002.

[64] S. Kinoshita, K. B. Cohen, P. V. Ogren, and L. Hunter. BioCreAtIvE Task1A: entity identification with a stochastic tagger. *BMC Bioinformatics*, 6 Suppl 1, 2005.

[65] K. Kipper, H. T. Dang, and M. S. Palmer. Class-Based Construction of a Verb Lexicon. In *Proceedings of AAAI/IAAI*, pages 691–696, 2000.

[66] S. Kok and P. Domingos. Extracting Semantic Networks from Text Via Relational Clustering. In *Proceedings of European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 624–639, 2008.

[67] M. Krallinger et al. The Protein-Protein Interaction tasks of BioCreative III: classification/ranking of articles and linking bio-ontology concepts to full text. *BMC Bioinformatics*, 12(S-8):S3, 2011.

[68] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, 1977.

[69] V. Law, C. Knox, Y. Djoumbou, T. Jewison, A. C. Guo, Y. Liu, A. Maciejewski, D. Arndt, M. Wilson, V. Neveu, A. Tang, G. Gabriel, C. Ly, S. Adamjee, Z. T. Dame, B. Han, Y. Zhou, and D. S. Wishart. DrugBank 4.0: shedding new light on drug metabolism. *Nucleic Acids Research*, 2013.

[70] D. Li, S. Somasundaran, and A. Chakraborty. ERD-MedLDA: Entity relation detection using supervised topic models with maximum margin learning. *Natural Language Engineering*, 18(2):263–289, 2012.

[71] D. Lin and P. Pantel. Discovery of Inference Rules for Question Answering. *Natural Language Engineering*, 7(4):343–360, 2001.

[72] T. Matsuzaki, Y. Miyao, and J. Tsujii. Efficient HPSG Parsing with Supertagging and CFG-Filtering. In *Proceedings of IJCAI*, pages 1671–1676, 2007.

[73] Mausam, M. Schmitz, S. Soderland, R. Bart, and O. Etzioni. Open Language Learning for Information Extraction. In *Proceedings of EMNLP-CoNLL*, pages 523–534, 2012.

[74] A. McCallum and W. Li. Early Results for Named Entity Recognition with Conditional Random Fields, Feature Induction and Web-enhanced Lexicons. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 188–191, 2003.

[75] R. McDonald, F. Pereira, S. Kulick, S. Winters, Y. Jin, and P. White. Simple Algorithms for Complex Relation Extraction with Applications to Biomedical IE. In *Proc. 43rd Annual Meeting of the Association for Computational Linguistics*, 2005.

[76] R. T. McDonald and F. Pereira. Identifying gene and protein mentions in text using conditional random fields. *BMC Bioinformatics*, 6(S-1), 2005.

[77] T. McIntosh, L. Yencken, J. R. Curran, and T. Baldwin. Relation Guided Bootstrapping of Semantic Lexicons. In *ACL (Short Papers)*, pages 266–270. The Association for Computer Linguistics, 2011.

[78] F. Mesquita, J. Schmidek, and D. Barbosa. Effectiveness and Efficiency of Open Relation Extraction. In *EMNLP*, pages 447–457. ACL, 2013.

[79] A. Mikheev, C. Grover, and M. Moens. Description of the LTG system used for MUC-7. In *In Proceedings of 7th Message Understanding Conference (MUC-7)*, 1998.

[80] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space. In *Proceedings of Workshop at ICLR*, 2013.

[81] T. Mikolov, Q. V. Le, and I. Sutskever. Exploiting Similarities among Languages for Machine Translation. *CoRR*, abs/1309.4168, 2013.

[82] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of the 27th Annual Conference on Neural Information Processing Systems*, pages 3111–3119, 2013.

[83] B. Min, S. Shi, R. Grishman, and C.-Y. Lin. Towards Large-Scale Unsupervised Relation Extraction from the Web. *International Journal on Semantic Web and Information System*, 8(3):1–23, 2012.

[84] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, 2009.

[85] T. Mitsumori, S. Fation, M. Murata, K. Doi, and H. Doi. Gene/protein name recognition based on support vector machine using dictionary as features. *BMC Bioinformatics*, 6(S-1), 2005.

[86] M. Miwa, T. Ohta, R. Rak, A. Rowley, D. B. Kell, S. Pyysalo, and S. Ananiadou. A method for integrating and ranking the evidence for biochemical pathways by mining reactions from text. *Bioinformatics*, 29(13):44–52, 2013.

[87] M. Miwa, R. Sætre, Y. Miyao, and J. Tsujii. Protein-protein interaction extraction by leveraging multiple kernels and parsers. *I. J. Medical Informatics*, 78(12):39–46, 2009.

[88] M. Miwa, P. Thompson, and S. Ananiadou. Boosting automatic event extraction from the literature using domain adaptation and coreference resolution. *Bioinformatics*, 28(13):1759–1765, 2012.

[89] Y. Miyao, T. Ohta, K. Masuda, Y. Tsuruoka, K. Yoshida, T. Ninomiya, and J. Tsujii. Semantic Retrieval for the Accurate Identification of Relational Concepts in Massive Textbases. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1017–1024, 2006.

[90] Y. Miyao, R. Sætre, K. Sagae, T. Matsuzaki, and J. Tsujii. Task-oriented Evaluation of Syntactic Parsers and Their Representations. In *Proceedings of ACL*, pages 46–54, 2008.

[91] A. Mnih and G. E. Hinton. A Scalable Hierarchical Distributed Language Model. In *Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems*, pages 1081–1088, 2008.

[92] A. A. Morgan, L. Hirschman, M. Colosimo, A. S. Yeh, and J. B. Colombe. Gene name identification and normalization using a model organism database. *Journal of Biomedical Informatics*, 37(6):396–410, Dec. 2004.

[93] F. Morin and Y. Bengio. Hierarchical probabilistic neural network language model. In *Proceedings of the International Workshop on Artificial Intelligence and Statistics*, pages 246–252, 2005.

[94] A. Moro and R. Navigli. Integrating syntactic and semantic analysis into the open information extraction paradigm. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, pages 2148–2154, 2013.

[95] D. Nadeau and S. Sekine. A survey of named entity recognition and classification. *Linguisticae Investigationes*, 30(1):3–26, January 2007. Publisher: John Benjamins Publishing Company.

[96] M. Narayanaswamy, K. E. Ravikumar, and K. Vijay-Shanker. A Biological Named Entity Recognizer. In *Pacific Symposium on Biocomputing*, 2003.

[97] V. Nebot and R. Berlanga. Exploiting semantic annotations for open information extraction: an experience in the biomedical domain. *Knowledge and Information Systems*, 38(2):385–369, 2014.

[98] C. Nédellec, R. Bossy, J.-D. Kim, J.-J. Kim, T. Ohta, S. Pyysalo, and P. Zweigenbaum. Overview of BioNLP Shared Task 2013. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 1–7, Aug. 2013.

[99] C. Nobata, P. Cotter, N. Okazaki, B. Rea, Y. Sasaki, Y. Tsuruoka, J. Tsujii, and S. Ananiadou. Kleio: a knowledge-enriched information retrieval system for biology. pages 787–788. ACM, 2008.

[100] E. W. Noreen. *Computer-Intensive Methods for Testing Hypotheses: An Introduction.* Wiley-Interscience, Hoboken, New Jersey, USA, 1989.

[101] T. Ohta, T. Matsuzaki, N. Okazaki, M. Miwa, R. Sætre, S. Pyysalo, and J. Tsujii. Medie and Info-pubmed: 2010 update. *BMC Bioinformatics*, 11(S-5):P7, 2010.

[102] P. Palaga, L. Nguyen, U. Leser, and J. Hakenberg. High-performance information extraction with AliBaba. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, volume 360, pages 1140–1143, 2009.

[103] H. Poon and P. Domingos. Joint Inference in Information Extraction. In *AAAI*, pages 913–918. AAAI Press, 2007.

[104] W. Pratt and M. Yetisgen-Yildiz. A Study of Biomedical Concept Identification: MetaMap vs. People. In *AMIA*, pages 529–533, 2003.

[105] S. Pyysalo, A. Airola, J. Heimonen, J. Bjorne, F. Ginter, and T. Salakoski. Comparative analysis of five protein-protein interaction corpora. *BMC Bioinformatics*, 9(Suppl 3):S6+, 2008.

[106] S. Pyysalo, A. Airola, J. Heimonen, J. Björne, F. Ginter, and T. Salakoski. Comparative analysis of five protein-protein interaction corpora. *BMC Bioinformatics*, 9(S-3), 2008.

[107] S. Pyysalo, F. Ginter, H. Moen, T. Salakoski, and S. Aniadou. Distributional Semantics Resources for Biomedical Text Mining. In *Proceedings of The 5th International Symposium on Languages in Biology and Medicine (LBM 2013)*, pages 39–43, 2013.

[108] S. Pyysalo, T. Ohta, and S. Ananiadou. Overview of the Cancer Genetics (CG) task of BioNLP Shared Task 2013. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 58–66, 2013.

[109] E. M. v. M. Quoc-Chinh Bui, David Campos and J. A. Kors. A fast rule-based approach for biomedical event extraction. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 104–108, Sofia, Bulgaria, August 2013. Association for Computational Linguistics.

[110] D. Rebholz-Schuhmann, A. Jimeno-Yepes, E. M. van Mulligen, N. Kang, J. Kors, D. Milward, P. Corbett, E. Buyko, E. Beisswanger, and U. Hahn. CALBC Silver Standard Corpus. *J. Bioinformatics and Computational Biology*, 8(1):163–179, 2010.

[111] D. Rebholz-Schuhmann, H. Kirsch, M. Arregui, S. Gaudan, M. Riethoven, and P. Stoehr. EBIMed - text crunching to gather facts for proteins from Medline. *Bioinformatics*, 23(2):237–244, 2007.

[112] M. Rei and T. Briscoe. Unsupervised Entailment Detection Between Dependency Graph Fragments. In *Proceedings of BioNLP 2011 Workshop*, pages 10–18, 2011.

[113] M. Richardson and P. Domingos. Markov Logic Networks. *Machine Learning*, 62:107–136, 2006.

[114] S. Riedel, H.-W. Chun, T. Takagi, and J. Tsujii. A Markov Logic Approach to Bio-Molecular Event Extraction. In *Proceedings of the BioNLP 2009 Workshop Companion Volume for Shared Task*, pages 41–49. Association for Computational Linguistics, June 2009.

[115] S. Riedel and A. McCallum. Robust biomedical event extraction with dual decomposition and minimal domain adaptation. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 46–50. Association for Computational Linguistics, 2011.

[116] S. Riedel, L. Yao, and A. McCallum. Modeling relations and their mentions without labeled text. In J. L. Balczar, F. Bonchi, A. Gionis, and M. Sebag, editors, *ECML/PKDD (3)*, volume 6323 of *Lecture Notes in Computer Science*, pages 148–163. Springer, 2010.

[117] L. Rimell, T. Lippincott, K. Verspoor, H. L. Johnson, and A. Korhonen. Acquisition and evaluation of verb subcategorization resources for biomedicine. *Journal of Biomedical Informatics*, 46(2):228–237, 2013.

[118] F. Rinaldi, G. Schneider, K. Kaljurand, M. Hess, and M. Romacker. An environment for relation mining over richly annotated corpora: the case of GENIA. *BMC Bioinformatics*, 7(S-3):S3, 2006.

[119] T. C. Rindflesch and M. Fiszman. The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text. *Journal of Biomedical Informatics*, 36(6):462–477, 2003.

[120] T. C. Rindflesch, H. Kilicoglu, M. Fiszman, G. Rosemblat, and D. Shin. Semantic MEDLINE: An advanced information management application for biomedicine. *Information Services & Use*, (31):15–21, 2011.

[121] T. C. Rindflesch, L. Tanabe, J. N. Weinstein, and L. Hunter. Edgar: Extraction of drugs, genes and relations from the biomedical literature. In *Pacific Symposium on Biocomputing*, pages 517–528, 2000.

[122] A. Ritter, Mausam, and O. Etzioni. A Latent Dirichlet Allocation Method for Selectional Preferences. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 424–434, 2010.

[123] R. Roller and M. Stevenson. Identication of Genia Events using Multiple Classifiers. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 125–129. Association for Computational Linguistics, 2013.

[124] G. Rosemblat, D. Shin, H. Kilicoglu, C. Sneiderman, and T. C. Rindflesch. A methodology for extending domain coverage in SemRep. *Journal of Biomedical Informatics*, 46(6):1099–1107, 2013.

[125] R. Sasano and S. Kurohashi. Japanese named entity recognition using structural natural language processing. In *Proceedings of IJCNLP 2008*, pages 607–612, 2008.

[126] D. O. Séaghdha. Latent Variable Models of Selectional Preference. In *ACL*, pages 435–444. The Association for Computer Linguistics, 2010.

[127] I. Segura-Bedmar, P. Martínez, and M. Herrero Zazo. SemEval-2013 task 9 : Extraction of Drug-Drug interactions from Biomedical Texts. In *Proceedings of SemEval 2013*, pages 341–350, June 2013.

[128] S. Sekine. Automatic paraphrase discovery based on context and keywords between NE pairs. In *In Proceedings of IWP*, 2005.

[129] B. Settles. ABNER: an open source tool for automatically tagging genes, proteins and other entity names in text. *Bioinformatics*, 21(14):3191–3192, 2005.

[130] P. K. Shah and P. Bork. LSAT: learning about alternative transcripts in MEDLINE. *Bioinformatics*, 22(7):857–865, 2006.

[131] R. Socher, E. H. Huang, J. Pennington, A. Y. Ng, and C. D. Manning. Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection. In *Proceedings of the 25th Annual Conference on Neural Information Processing Systems*, pages 801–809, 2011.

[132] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning. Semi-supervised Recursive Autoencoders for Predicting Sentiment Distributions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 151–161, 2011.

[133] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: A Core of Semantic Knowledge. In *Proceedings of the 16th international World Wide Web conference (WWW 2007)*, pages 697–706, 2007.

[134] A. Sun, R. Grishman, and S. Sekine. Semi-supervised Relation Extraction with Large-scale Word Clustering. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 521–529.

[135] L. Sun and A. Korhonen. Improving Verb Clustering with Automatically Acquired Selectional Preferences. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 638–647, 2009.

[136] M. Surdeanu, S. Harabagiu, J. Williams, and P. Aarseth. Using Predicate-Argument Structures for Information Extraction. In *Proceedings of ACL*, pages 8–15. Association for Computational Linguistics.

[137] A. Tamura, T. Watanabe, and E. Sumita. Recurrent Neural Networks for Word Alignment Model. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1470–1480, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[138] L. K. Tanabe and W. J. Wilbur. Tagging gene and protein names in biomedical text. *Bioinformatics*, 18(8):1124–1132, 2002.

[139] Y. Tateisi, T. Ohta, and J. ichi Tsujii. Annotation of Predicate-argument Structure on Molecular Biology Text. In *In Proceedings of the Workshop on the 1st International Joint Conference on Natural Language Processing (IJCNLP*, 2004.

[140] K. Taura, T. Matsuzaki, M. Miwa, Y. Kamoshida, D. Yokoyama, N. Dun, T. Shibata, C. S. Jun, and J. Tsujii. Design and Implementation of GXP Make - A Workflow System Based on Make. In *eScience*, pages 214–221. IEEE Computer Society, 2010.

[141] P. Thompson, J. McNaught, S. Montemagni, N. Calzolari, R. Del Gratta, V. Lee, et al. The BioLexicon: a large-scale terminological resource for biomedical text mining. *BMC Bioinformatics*, 12(1):397, 2011.

[142] E. F. Tjong Kim Sang and F. De Meulder. Introduction to the CoNLL-2003 Shared Task: Language-independent Named Entity Recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 142–147, 2003.

[143] R. Tsai, W.-C. Chou, Y.-S. Su, Y.-C. Lin, C.-L. Sung, H.-J. Dai, I. Yeh, W. Ku, T.-Y. Sung, and W.-L. Hsu. BIOSMILE: A semantic role labeling system for biomedical verbs using a maximum-entropy model with automatically generated template features. *BMC Bioinformatics*, 8(1):325, 2007.

[144] Y. Tsuruoka, M. Miwa, K. Hamamoto, J. Tsujii, and S. Ananiadou. Discovering and visualizing indirect associations between biomedical concepts. *Bioinformatics [ISMB/ECCB]*, 27(13):111–119, 2011.

[145] Y. Tsuruoka and J. Tsujii. Improving the performance of dictionary-based approaches in protein name recognition. *Journal of Biomedical Informatics*, 37(6):461–470, 2004.

[146] S. Van Landeghem, J. Björne, C.-H. Wei, K. Hakala, S. Pyysalo, S. Ananiadou, H.-Y. Kao, Z. Lu, T. Salakoski, Y. Van de Peer, and F. Ginter. Large-scale event extraction from literature with multi-level gene normalization. *PLoS One*, 8(4), 2013.

[147] M. Vazquez, M. Krallinger, F. Leitner, and A. Valencia. Text mining for drugs and chemical compounds: Methods, tools and applications. *Mol. Inf.*, 30(6-7):506–519, June 2011.

[148] H. M. Wallach, I. Murray, R. Salakhutdinov, and D. Mimno. Evaluation Methods for Topic Models. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1105–1112, 2009.

[149] H. Wang, Y. Ding, J. Tang, X. Dong, B. He, J. Qiu, and D. J. Wild. Finding Complex Biological Relationships in Recent PubMed Articles Using Bio-LDA. *PLoS One*, 3(6), 2011.

[150] T. Wattarujeekrit, P. K. Shah, and N. Collier. PASBio: predicate-argument structures for event extraction in molecular biology. *BMC Bioinformatics*, 5:155, 2004.

[151] F. Wu and D. S. Weld. Open Information Extraction Using Wikipedia. In J. Hajic, S. Carberry, and S. Clark, editors, *ACL*, pages 118–127. The Association for Computer Linguistics, 2010.

[152] A. B. Xiao Liu and Y. Grandvalet. Biomedical Event Extraction by Multi-class Classification of Pairs of Text Entities. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 45–49, Sofia, Bulgaria, August 2013. Association for Computational Linguistics.

[153] R. Xu and Q. Wang. Large-scale extraction of accurate drug-disease treatment pairs from biomedical literature for drug repurposing. *BMC Bioinformatics*, 14:181, 2013.

[154] Y. Xu, M.-Y. Kim, K. Quinn, R. Goebel, and D. Barbosa. Open information extraction with tree kernels. In *Proceedings of NAACL-HLT 2013*, pages 868–877, Atlanta, Georgia, June 2013.

[155] A. Yakushiji, Y. Miyao, T. Ohta, Y. Tateisi, and J. Tsujii. Automatic construction of predicate-argument structure patterns for biomedical information extraction. In *Proceedings of EMNLP*, pages 284–292, 2006.

[156] A. Yakushiji, Y. Miyao, Y. Tateisi, and J. Tsujii. Biomedical information extraction with predicate-argument structure patterns. In *The First International Symposium on Semantic Mining in Biomedicine*, pages 60–69, 2005.

[157] L. Yao, A. Haghighi, S. Riedel, and A. McCallum. Structured Relation Discovery using Generative Models. In *EMNLP*, pages 1456–1466, 2011.

[158] L. Yao, S. Riedel, and A. McCallum. Unsupervised Relation Discovery with Sense Disambiguation. In *ACL (1)*, pages 712–720, 2012.

[159] A. Yates and O. Etzioni. Unsupervised methods for determining object and relation synonyms on the web. *Journal of Artifitical Intelligence Research*, 34(1):255–296, Mar. 2009.

[160] A. Yeh, A. Morgan, M. Colosimo, and L. Hirschman. BioCreative task 1A: Gene mention finding evaluation. *BMC Bioinformatics 2005*, 6(1):S2, 2005.

[161] K. Yoshida and J. Tsujii. Reranking for biomedical named-entity recognition. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*, BioNLP '07, pages 209–216. Association for Computational Linguistics, 2007.

[162] S. Zhang and N. Elhadad. Unsupervised biomedical named entity recognition: Experiments with clinical and biological texts. *Journal of Biomedical Informatics*, 46(6):1088–1098, 2013.

[163] S. Zhao. Named Entity Recognition in Biomedical Texts Using an HMM Model. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and Its Applications*, JNLPBA '04, pages 84–87. Association for Computational Linguistics, 2004.

[164] G. Zhou and J. Su. Exploring Deep Knowledge Resources in Biomedical Name Recognition. In *COLING 2004 International Joint workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP) 2004*, pages 99–102. COLING, August 28th and 29th 2004.

[165] J. Zhu, Z. Nie, X. Liu, B. Zhang, and J.-R. Wen. StatSnowball: a statistical approach to extracting entity relationships. In *WWW '09: Proceedings of The 18th International Conference on World Wide Web*, pages 101–110. ACM, 2009.

[166] W. Y. Zou, R. Socher, D. M. Cer, and C. D. Manning. Bilingual Word Embeddings for Phrase-Based Machine Translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1393–1398, 2013.

Publications

[1] Nhung T. H. Nguyen, Makoto Miwa, Yoshimasa Tsuruoka, Takashi Chikayama and Satoshi Tojo: “Wide-Coverage Relation Extraction from MEDLINE Using Deep Syntax”, BMC Bioinformatics (in press) (December 2014).

[2] Nhung T. H. Nguyen, Makoto Miwa, Yoshimasa Tsuruoka, and Satoshi Tojo: “Identifying synonymy between relational phrases using word embeddings”, Journal of Biomedical Informatics (conditionally accepted) (December 2014).

[3] Nhung T. H. Nguyen, Makoto Miwa, Yoshimasa Tsuruoka and Satoshi Tojo: “Open Information Extraction from Biomedical Literature Using Predicate-Argument Structure Patterns”, Proceedings of The 5th International Symposium on Languages in Biology and Medicine (LBM 2013), pp. 51-55 (December 2013).

[4] Nhung T. H. Nguyen and Yoshimasa Tsuruoka: “Extracting Bacteria Biotopes with Semi-supervised Named Entity Recognition and Coreference Resolution”, Proceedings of The BioNLP 2011 Workshop Companion Volume for Shared Task, pp. 94-101, Association for Computational Linguistics (June 2011).

[5] Nhung T. H. Nguyen, Vinh Q. Le, Quoc-Minh Nghiem and Dien Dinh: “A General Approach for Word Reordering in English-Vietnamese-English Statistical Machine Translation”, International Journal on Artificial Intelligent Tools, World Scientific (under second round of review) (August 2014).

[6] Vu C. D. Hoang, Aw Ai Ti and Nhung T. H. Nguyen: “A Rule-Augmented Statistical Phrase-based Translation System”, Proceedings of The 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014): System Demonstrations, pp. 73-78 (June 2014).

[7] Thach V. Bui, Oanh K. Nguyen, Van H. Dang, Nhung T. H. Nguyen and Thuc D. Nguyen: “A Variant of Non-Adaptive Group Testing and Its Application in Pay-

Television via Internet", Proceedings of Information and Communication Technology - International Conference, ICT-EurAsia 2013, pp. 324-330, Lecture Notes in Computer Science, Volume 7804 (January 2013).

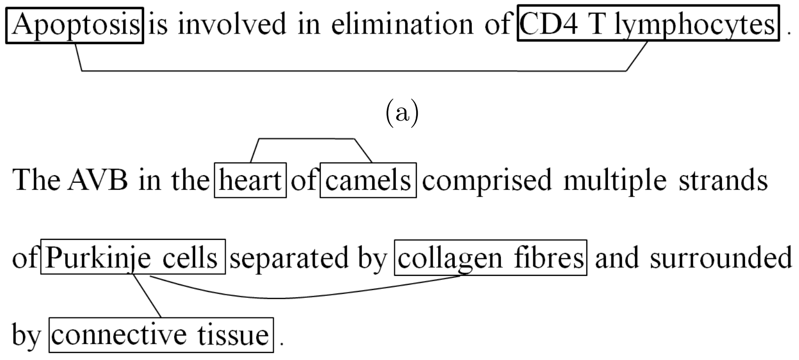
[8] Hien M. Vo, Dien Dinh and Nhung T. H. Nguyen, "A Fast Decoder Using Less Memory", Proceedings of The 4th International Conference on Knowledge and Systems Engineering, pp. 173-180, IEEE (August 2012).

Appendix A

Guideline for Manual Evaluation

A.1 Biomedical binary relations

In our scenario, a biomedical binary relation is composed by two biomedical entities to show associations or effects between the entities. For instance, in Figure A.1a, there is one binary relation between ‘Apoptosis’ and ‘CD4 T lymphocytes’, which indicates that ‘Apoptosis’ somehow affects ‘CD4 T lymphocytes’. Figure A.1b presents three relations r1(heart, camels), r2(Purkinje cells, collagen fibres) and r3(Purkinje cells, connective tissue), these relations tell us that there are associations between these entities. The associations can be of any type, such as ‘part-of’, ‘separated by’, and ‘surrounded by’ relations.



Apoptosis is involved in elimination of CD4 T lymphocytes .

(a)

The AVB in the heart of camels comprised multiple strands of Purkinje cells separated by collagen fibres and surrounded by connective tissue .

(b)

Figure A.1: Examples of biomedical binary relations.

A.2 Evaluation criteria

We define two evaluation criteria for entities and relations.

A.2.1 Evaluating entities

Entities in our setting are nouns or base noun phrases in a sentence. An entity is correct if and only if its content words represent the most complete meaning within the sentence containing it.

Example 1: Alterations in the microcirculatory bed of the thalamus resulting from thermal trauma ...

Entity	**Correct?**	**Comments**
microcirculatory bed	Yes	
thalamus	Yes	
trauma	No	It should be 'thermal trauma'

In example 1, the first two entities 'microcirculatory bed' and 'thalamus' are correct, but the entity 'trauma' is NOT. The reason is that trauma' does not reflect the complete meaning that this sentence aims at; the right one should be thermal trauma'.

Follows are some rules applied to some specific cases.

Rule 1 for discontinuous entities

- It should be noted that in biomedical text, sometimes, entities appear in *discontinuous* text regions. For instance, given the following sentence:

  Example 1a: We investigated spontaneous and lipopolysaccharide (**LPS**) stimulated production of **tumor necrosis factor alpha** (TNF alpha), **interleukin** (IL) **1**, **IL-6**, and **IL-8** .

Entity	**Correct?**	**Comments**
LPS	Yes	
tumor necrosis factor alpha	Yes	
interleukin 1	Yes	Correct even though it is discontinuous.
IL-6	Yes	
IL-8	Yes	

In this sentence, the entity 'interleukin 1' is correct even though it is discontinuous.

- There are cases in which discontinuous entities are not correct, such as entity 'interventricular septum' in example 1b. The entity is not correct, it should be 'interventricular membranous septum'.

  Example 1b: The atrioventricular bundle entered the lower part of the interventricular membranous septum ...

Entity	Correct?	Comments
atrioventricular bundle	Yes	
interventricular septum	No	It should be 'interventricular membranous septum'.

Rule 2 for noun modifiers

- An entity is *correct* even if it fails to include the common nouns or head nouns located at the end of the phrase, as long as the entity conveys the main meaning of the phrase. For instance, entities 'probiotic' in example 2a is correct since its meaning is sufficient without including the word 'effects'. More specifically, in the noun phrase 'probiotic effects', 'effects' is the head noun and modified by 'probiotic', which is a bacteria name that expresses the main meaning of this phrase. Therefore, 'probiotic' can be correct in our setting. The same explanation applies to 'ciguatera' in example 2b.

  Example 2a: Saccharomyces boulardii is a strain of yeast which has been extensively studied for its probiotic effects.

Entity	Correct?	Comments
Saccharomyces boulardii	Yes	
probiotic	Yes	'probiotic' is equal to 'probiotic effects'
yeast	Yes	

  Example 2b: Ciguatoxins, the principal causative toxins of ciguatera seafood poisoning, are large ladder-like polycyclic ethers.

- If the extracted entities are common nouns or head nouns, they are not correct. For example, 'progastrin' in example 2c is the head of noun phrase 'tissue progastrin' and modified by 'tissue'. It is not correct since in this context it should include the modifier 'tissue' to be specific enough.

  Example 2c: ... tissue progastrin was elevated by only about 50%.

Entity	**Correct?**	**Comments**
Ciguatoxins	Yes	
toxins	Yes**	According to rule 3
ciguatera	Yes*	It is correct since ‘ciguatera’ itself also includes the meaning of ‘seafood poisoning’.
poisoning	No	‘ciguatera seafood poisoning’
ethers	No	It should be ‘large ladder-like polycyclic ethers’ or ‘ladder-like polycyclic ethers’.

Entity	**Correct?**	**Comments**
progastrin	No	It should be ‘tissue progastrin’.

Rule 3 for adjective modifiers

- Ideally, adjectives/adjective phrases that modify nouns/noun phrases should be included in the extracted entities. However, if the adjectives/adjective phrases are general ones, such as ‘large’, ‘excessive’, ‘principal’, and ‘causative’, they can be excluded from the entities. For instance, entity ‘toxins’ in example 2b is correct even though it does not include the adjective phrase ‘principal causative’. Entity ‘selenium’ in example 4c is also correct without the adjective ‘excessive’.

- By contrast, if that adjective or adjective phrase presents a *biological* meaning, it must be included in the entity, such as in example 1, the adjective ‘thermal’ must be included in ‘thermal trauma’ to make its meaning complete. This rule is demonstrated in the following examples.

  Example 3a: The atrioventricular bundle ran through the *fibrous* trigone ...

Entity	**Correct?**	**Comments**
atrioventricular bundle	Yes	
trigone	No	It should be ‘fibrous trigone’.

Example 3b: ... was investigated in thirty *prepubertal* children.

Entity	**Correct?**	**Comments**
children	No	It should be ‘prepubertal children’.

Example 3c: Laminin is located in the zone of the basal membrane.

Entity	**Correct?**	**Comments**
Laminin	Yes	
membrane	No	It should be ‘basal membrane’.

Rule 4 for possessive forms

- If there is a preposition ‘of’ between two entities to show their part-whole relation, and these two entities have sufficient meaning, they are correct. For example, entities ‘heart’ and ‘camels’ in the following sentence are correct, even though the proper entity should be ‘heart of camels’. The same explanation applies to ‘strands’ and ‘Purkinje cell’.

  Example 4a: the AVB in the heart of camels comprised multiple strands of Purkinje cells ...

Entity	**Correct?**	**Comments**
heart	Yes	Even though the proper one is ‘heart of camels’.
camels	Yes	Even though the proper one is ‘heart of camels’.
strands	Yes	Even though the proper one is ‘strands of Purkinje cells’.
Purkinje cells	Yes	Even though the proper one is ’strands of Purkinje cells’.

  Example 4b: Responses of **rhesus monkeys** were reinforced by delivery of either a **pentobarbital** (4.0 mg/ml) *solution* or a vehicle (water) or **saccharin** *solution* under a concurrent signaled differential reinforcement of low rates 30-s schedule.

Entity	**Correct?**	**Comments**
rhesus monkeys	Yes*	‘responses of rhesus monkeys’
pentobarbital	Yes	According to rule 2a
saccharin	Yes	According to rule 2a

  Example 4c: An excessive selenium supply compensated to a great extent for the effects of vitamin E deficiency on IgG and IgA.

Entity	**Correct?**	**Comments**
selenium	Yes	According to rule 3
vitamin E deficiency	Yes*	’effects of vitamin E deficiency’
IgA	Yes	

- Strictly speaking, in example 4b, the entity ‘rhesus monkeys’ should be ‘responses of rhesus monkeys’, which is more specific and accurate in this context. However, in our setting, ‘rhesus monkeys’ is acceptable because we can infer from the sentence that somehow there is a vague relation between ‘rhesus monkeys’ and ‘pentobarbital’, and we would like to extract such vague relations also. Entity ‘vitamin E deficiency’ in example 4c also demonstrates this exception.

A.2.2 Evaluating extracted relations

A correct relation must satisfy the following two conditions:

- The two entities composing the relation must be correct according to the above-mentioned criteria.

- The semantic relationship between two entities in the relation must be represented explicitly by some linguistic expression.

Any relations that break one of the above two conditions are incorrect.

For example, all extracted relations in Figure A.1 are correct since they satisfy our criteria: (1) all extracted entities are correct; (2) their semantic relationship are presented explicitly such as “[Apoptosis] ... involved in ... [CD4 T lymphocytes]”; “... [Purkinje cells] separated by [collagen fibres] ...”

By contrast, all extracted relations in Figure A.2 are not correct. The two relations r1 and r4 break condition 1 since the entities ‘membrane’ and ‘vessels’ are not correct. Relation r2 breaks condition 2 because this sentence has two clauses: one is about ‘Laminin’ and the other is about ‘tenascin’, and there is no information to show their semantic relationship. Relation r3 breaks both conditions because entity ‘vessels’ is not correct and the relationship between ‘Laminin’ and ‘vessels’ is not presented in this sentence.

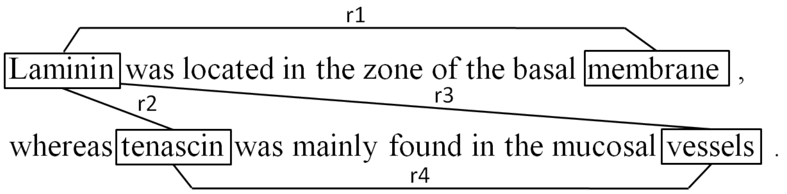
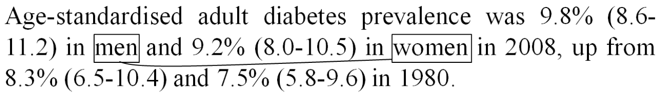
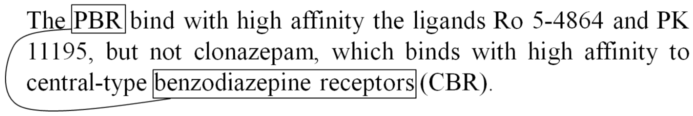
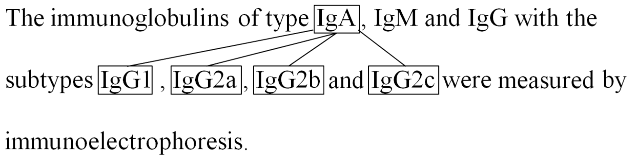
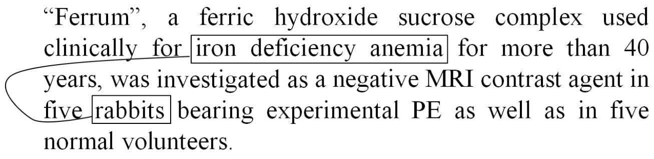
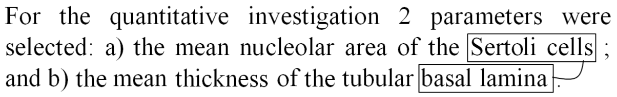

r1
Laminin was located in the zone of the basal membrane ,
r2
r3
whereas tenascin was mainly found in the mucosal vessels .
r4

Figure A.2: Examples of extracted relations that do not satisfy the two evaluation principles. r1 and r4 break principle 1; r2 breaks principle 2; r3 breaks both principle.

For illustrating condition 2 clearly, we have listed more examples of extracted relations that break this condition in Table A.1.

Table A.1: Examples of extracted relations that are incorrect because their semantic relationships are not shown in the sentence.

Extracted relations	**Evaluation**
 Age-standardised adult diabetes prevalence was 9.8% (8.6-11.2) in men and 9.2% (8.0-10.5) in women in 2008, up from 8.3% (6.5-10.4) and 7.5% (5.8-9.6) in 1980.	The relation between 'men' and 'women' is incorrect since their semantic relationship is not mentioned in this sentence.
 The PBR bind with high affinity the ligands Ro 5-4864 and PK 11195, but not clonazepam, which binds with high affinity to central-type benzodiazepine receptors (CBR).	This extracted relation is incorrect because of two reasons. Firstly, it breaks condition 1 since entity 'benzodiazepine receptors' is incorrect. Secondly, this sentence discusses two independent topics, one is 'PBR' and the other is 'clonazepam'.
 The immunoglobulins of type IgA, IgM and IgG with the subtypes IgG1 , IgG2a, IgG2b and IgG2c were measured by immunoelectrophoresis.	All four extracted relation are incorrect because they breaks condition 2. We can see that 'IgG1', 'IgG2a', 'IgG2b' and 'IgG2c' are subtypes of 'IgG', and this sentence lists 'IgA' and 'IgG' but says nothing about their relation. Therefore, there is no relationship between 'IgA' and 'IgG' subtypes.
 "Ferrum", a ferric hydroxide sucrose complex used clinically for iron deficiency anemia for more than 40 years, was investigated as a negative MRI contrast agent in five rabbits bearing experimental PE as well as in five normal volunteers.	The relation between 'iron deficiency anemia' and 'rabbits' is incorrect. We can infer from this sentence that 'Ferrum' was used for two independent purposes. One is related to 'iron deficiency anemia', and the other is related to 'negative MRI contrast agent in rabbits'. However, this sentence does not mention the relationship between these purposes.
 For the quantitative investigation 2 parameters were selected: a) the mean nucleolar area of the Sertoli cells ; and b) the mean thickness of the tubular basal lamina .	This relation is incorrect since it breaks both conditions. Firstly, entity 'basal lamina' is not correct, it should be 'tubular basal lamina'. Secondly, this sentence lists two selected parameters that are related to 'Sertoli cells' and 'tubular basal lamina', but no relationship between them are mentioned.

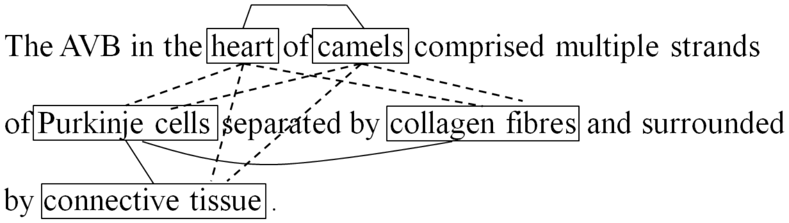

The AVB in the heart of camels comprised multiple strands
of Purkinje cells separated by collagen fibres and surrounded
by connective tissue .

Figure A.3: An example of indirect relations (dash lines). These relations are not directly represented through the syntactic structure but can be inferred based on syntactic clues.

Exception 1

There are some cases where the relation between two entities is not directly shown by the syntactic structure, but if that relation can be inferred through the sentence, it can be assessed as a TRUE relation. The example in Figure A.3 illustrates this case.

The system extracts nine relations; three of them, represented by solid lines, are correct since we can see the syntactic clues very clearly. The relation between 'heart' and 'Purkinje cells', represented by a dash line, is inferred based on the following reasoning: 'the AVB' is a part of the 'heart', 'the AVB' comprises 'strands of Purkinje cells', therefore 'heart' and 'Purkinje cells' most likely have some relations. The other five indirectly relations can be inferred in the same way.

A.3 The output's format

A test set including 500 sentences randomly selected from MEDLINE was given to four different systems. These systems returned a set of binary relations as output. Each binary relation is presented in four fields consisting of (1) the start position of the first entity, (2) the first entity, (3) the start position of the second entity and (4) the second entity in a sentence, as shown in Table A.2.

A.4 Tasks for annotators

The annotators are required to:

- Evaluate all binary relations extracted from the 500 sentences by the four systems.
- Strictly follow our guideline to assess the extracted relations:
  - Extracted relations that satisfy the two conditions are TRUE,

Table A.2: Samples of the output and the evaluation of binary relations, the two final columns are filled by annotators.

Start1	Entity 1	Start2	Entity 2	TRUE/ FALSE	Comments
Sentence 1: The atrioventricular bundle ran through the fibrous trigone and entered the lower part of the interventricular membranous septum, beneath the right endocardium, then lay over or slightly to the side of the centre of the muscular interventricular crest.					
4	atrioventricular bundle	52	trigone	FALSE	P1
4	atrioventricular bundle	148	endocardium	FALSE	P1
4	atrioventricular bundle	4	interventricular septum	FALSE	P1
4	atrioventricular bundle	246	crest	FALSE	P1
Sentence 2 :The detection of the illegal use of clenbuterol (CBL) as a growth promoter has relied on detecting residual concentrations of the drug in body fluids or tissues.					
36	clenbuterol	138	body fluids	TRUE	
36	clenbuterol	153	tissues	TRUE	

- otherwise they are FALSE.

When the evaluators assign FALSE to a relation, please specify which condition is not satisfied. If it breaks the first condition, please write ‘P1’ in the column ‘Comments’. If it breaks the second one, please write ‘P2’. If it breaks both, please write ‘both’. Otherwise, please tell us your opinion. In case of exception 1 in section 2.2, if the annotators assign FALSE to indirect relations and the reason is not P1, they have to explain their reasoning clearly. In case the annotators do not follow any rules or principles, please clarify the reason.