# EXTRACTION OF KNOWLEDGE FOR MICRORNAS AND GENES: EXTRACTING CONNECTIONS THROUGH ASSOCIATION, INVOLVEMENT, AND REGULATION

by

Samir Gupta

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

Spring 2021

© 2021 Samir Gupta
All Rights Reserved

**EXTRACTION OF KNOWLEDGE FOR MICRORNAS AND GENES: EXTRACTING CONNECTIONS THROUGH ASSOCIATION, INVOLVEMENT, AND REGULATION**

by

Samir Gupta

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

Approved: _____
Levi T. Thompson, Ph.D.
Dean of the College of Engineering

Approved: _____
Louis F. Rossi, Ph.D.
Vice Provost for Graduate and Professional Education and Dean of the Graduate College

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

Signed: _____

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

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

Signed: _____

Cathy H. Wu, Ph.D.
Member of dissertation committee

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

Signed: _____

Kathleen F. McCoy, Ph.D.
Member of dissertation committee

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

Signed: _____

Andrew Su, Ph.D.
Member of dissertation committee

## ACKNOWLEDGMENTS

This dissertation would not have been possible without the guidance and support of several individuals.

Foremost, I would like to express my sincere gratitude to my advisor, Dr. Vijay K. Shanker for his support and mentorship during my Ph.D. study. His continued encouragement, guidance, and feedback through the countless meetings and discussions have played an integral role in shaping this dissertation. Dr. Shanker taught me how to critically think and approach research problems as well as how to disseminate my research through research papers and presentations. These skills have helped me in my research, and I will continue to benefit from them in my future career. Most importantly, he has shown me, by his example, what a good person (and researcher) should be.

I would also like to thank the rest of my dissertation committee members: Dr. Cathy H. Wu, Dr. Kathleen F. McCoy, and Dr. Andrew Su for taking the time out of their busy schedules and serving on my committee. I thank them for their insightful comments, suggestions, and hard questions that enabled me to improve my dissertation.

Many thanks to my fellow lab mates at the BioNLP and Text-mining Group: Yifan Peng, Ruoyao Ding, Gang Li, and Ashique Mahmood, Peng Su, and Debarati Roychowdhury for being excellent colleagues and being always available to discuss ideas. During my many years at UD, I have made some lifelong, amazing friends, naming a few, in no particular order: Abhishek Kolagunda, Ayush Dusia, Diganto

Ghosh, Siddhisanket Raskar, Pradnya Powar, Karankumar Sabhnani, and Prathamesh Kharkar. I will always be grateful for their friendship and the memories we shared.

Lastly, and most importantly, I would like to thank my parents – Kunti Gupta and Mohan Prasad Gupta, and my brother – Shekhar Gupta, whose unconditional love and support are always with me in whatever I pursue. I dedicate this dissertation to my mom, Kunti Gupta, for her unending faith in me and for providing me with inspiration, encouragement, and strength at every step in my life.

## TABLE OF CONTENTS

Appendix

## LIST OF TABLES

## LIST OF FIGURES

## ABSTRACT

Biological entities such as genes, proteins, and microRNAs are critical players in various biological processes and diseases. The role of these entities on biological processes and diseases forms a significant part of biomedical knowledge bases. However, a large portion of this information is buried in scientific literature as unstructured text. This work is motivated by our belief that the development of relation extraction systems that capture the roles of such entities on different processes and diseases from literature is important and much needed. We hypothesize that connections between biological entities and concepts as stated in text can be captured by the extraction of a small number of relations, which we call CAIR relations: Connections through Association, Involvement, and Regulation.

We have developed a general relation extraction framework that reduces the effort required for developing individual relation extraction (RE) systems. This framework is based on a structured representation called Extended Dependency Graph (EDG), which utilizes syntactic dependencies and information beyond syntax to capture thematic dependencies. Based on this framework, we developed a general CAIR relation extraction system to connect a bio-entity to associated concepts. To demonstrate the wide applicability of CAIR relations and the framework, we have developed several RE systems and text-mining applications including miRiaD, a tool to extract the role of microRNAs in diseases, and Phos2X, a tool to extract the functional impact of protein phosphorylation. Additionally, as a continuation of miRiaD development, we also developed DEXTER, a tool to extract microRNA's

differential expression level information in diseases, which covers a different aspect of microRNA-disease associations. Such differential expression statements are stated through comparative sentences, comparing expression levels in two different samples. Thus, we have developed a general system to identify comparison sentences and extract the various components (compared aspect, compared entities/scenarios, and scale of the comparison). Additionally, we extended DEXTER to also extract gene expression information. All the tools we have developed have been evaluated by comparing with human annotations and show high precision and recall.

# Chapter 1

# INTRODUCTION

## 1.1 Motivation

Proteins are large complex molecules that play many critical roles in cell biology. They are involved in cellular processes and are required for the structure, function, and regulation of tissues and organs. Genes contain the information needed to create proteins. Transcription and translation are two major steps that convert the information in genes to make proteins. MicroRNAs are a class of small non-coding RNAs encoded in the genomes of animals, plants, and protozoa that affect gene regulation. In general, microRNAs negatively regulate protein levels that either inhibit their translation or triggers their cleavage. The genes, the corresponding proteins, and microRNAs are entities that play a major role in a cell and have a significant influence on other entities and the cell itself.

To understand the impact of microRNAs and genes on cell biology, it is important to know how these entities are related to their environment. A lot of attention in biomedical text mining has been paid on extracting relations between different biological entities such as protein-protein interactions (PPI) [1–4], post-translational modifications (PTM) [5–8], mutation-gene associations [9–12], subcellular localization [13–16], chemical protein interactions [17–19]. These interactions at a cellular level have an impact at a higher level on different biological processes, which can involve hundreds of these interactions. The role of microRNAs

and genes on biological processes and disease (aberrant process) forms a significant part of biomedical knowledge bases. However, a large portion of this information is buried in scientific literature as unstructured text. Extraction of the different roles of such entities in processes and diseases from literature is important for quick understanding and hypothesis generation and is the focus of this dissertation. Most of the existing relation extraction (RE) tools in the biomedical domain focus on genes/proteins and specific types of relations such as PPI, PTM relations. Extraction of information about microRNA's association with diseases and biological processes from text are limited. This motivated my initial dissertation work to focus on microRNAs.

Most of the work for microRNAs is focused on the extraction of its regulation of genes. However, a literature survey suggested that there is also an abundance of papers discussing microRNAs and their role in processes and diseases. Several databases capture microRNA-disease associations such as miR2Disease [20], miRCancer [21], and the Human microRNA Disease Database (HMDD) [22]. miR2Disease and miRCancer provide information on microRNA expression in disease, and miR2Disease additionally covers microRNA target genes in the context of their impact on disease. All these databases involve manual curation to associate microRNA information from scientific literature; thus they are limited by the time-consuming nature of manual curation and have difficulty keeping up with the explosion of publications in the biomedical field. Figure 1.1 shows the trend in the numbers of papers published from the year 2000 to 2019 obtained by a PubMed search for the query 'microRNA'. Thus, there is a need for automated literature mining tools to streamline and accelerate the curation process as well as provide researchers with a

general resource containing microRNA's relationships for fast access to the most recent and relevant published information.

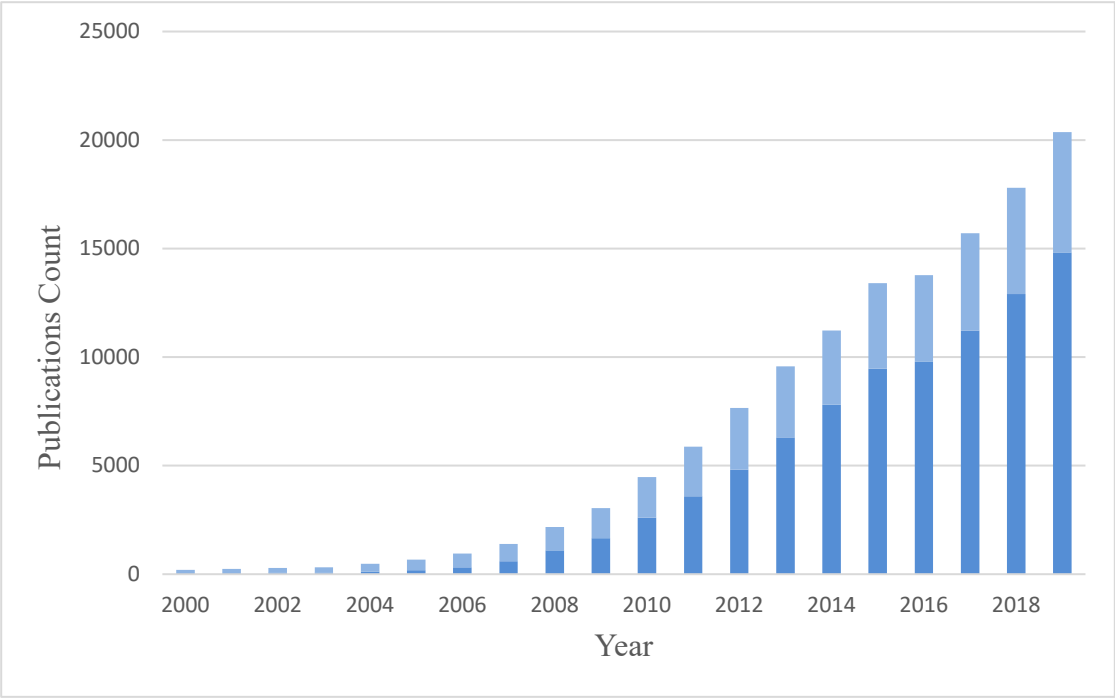


Figure 1.1: Number of microRNA publications from 2000 to 2019 in PubMed. (dark blue indicates the number of microRNA publications, where a disease is also mentioned)

To address this need for text-mining (TM) tools, we developed **miRiaD** [23], a tool that can extract associations between microRNAs and diseases from the literature. During the development of this tool, we noticed that nearly all statements connecting microRNAs to processes and disease-related terms were through mentions of regulation relation (mir-31 "regulates/promotes/inhibits" apoptosis) as well as involvement (mir-31 "is involved in" apoptosis) and association (mir-31 "was

associated with" apoptosis). We termed these relations as **CAIR** relations (Connections through **A**ssociation, **I**nvolvement, and **R**egulation). Subsequent to the development of miRiaD, I started developing a RE system to extract such CAIR relations based on the hypothesis that such relations are widely used in connecting biomolecular entities with diseases and processes, which are important for microRNAs' connection to diseases.

We believe that the statements expressing CAIR relations are widely used and therefore important to capture. To demonstrate the generality of these relations and that their extraction generalizes beyond the miRiaD task, we developed a system called **Phos2X** that extracts the impact of protein phosphorylation on different properties of the phosphorylated protein (the substrate). Specifically, we consider the effect of phosphorylation on the substrate's protein-protein interactions, its post-translational modifications, and subcellular localizations, which are typically stated in text through CAIR relation.

While developing miRiaD, as a tool to capture the role of microRNAs (miRs) in disease, we noticed that the importance of microRNAs in disease is often first noted as their over-/under-expression of in disease sample. While such statements might not necessarily suggest that the miR has a role in the disease, we believe they are important to capture. Thus, we developed a tool called **DEXTER** [24], which extracts text evidence of genes/microRNAs expression level in diseases. A major proportion of statements regarding the differential expression of genes/microRNA in disease use comparative statements, comparing expression levels in two different samples. Thus work on extracting comparison relations was critical to the development of DEXTER.

The relevance of comparison statements goes beyond statements of differential expression of genes and microRNAs. Biomedical researchers conduct experiments to validate their hypotheses and infer associations making observations under two different scenarios (e.g., disease sample vs. control sample). When the differences between the groups are statistically significant, the association can be inferred. Thus we developed a **RE system to identify comparison** [25] sentences and also extract the various components of a comparison relation (compared aspect, compared entities, and scale of the comparison) as an extension to the CAIR systems. The next subsections summarize the different parts of my dissertation work and the contributions they make.

## 1.2 A Framework for Relation Extraction

Since we are interested in the extraction of a wide range of relations, I was motivated in building a general framework that reduces the effort of developing individual relation extraction (RE) systems by handling certain common aspects of such RE systems, such as parsing, syntactic dependency conversion, argument propagation and developing a template for writing rules. The ultimate goal of this general framework is that any user developing a new RE system will simply have to state lexico-syntactic patterns at a level of generalization provided by syntactic dependencies.

Thus we developed a relation extraction (RE) framework, which is based on the Extended Dependency Graph (EDG) [3]. Earlier, we had proposed this new representation EDG [3] to account for and unify the various syntactic variations (active, passive, nominalized, etc.) and move towards more thematic dependencies (numbered-argument). EDG unifies different syntactic variations and provides

consistent labeling across them using general linguistic principles. We have since extended EDG in this dissertation to develop a general biomedical relation extraction (RE) framework that will be used to extract relations of interest and can also be used for the extraction of other relations as well. In Chapter 2, we describe this relation extraction (RE) framework based on the ideas of EDG. Based on this relation extraction framework, we have developed general RE systems to extract CAIR and comparison relations that are widely used in connecting biomolecular entities with processes and diseases.

### 1.2.1 CAIR relations

The development of TM tools to extract information about the role of microRNAs on cell biology was a primary focus of my work. At a mechanistic level microRNAs influence biological processes and diseases through their effects on the expression of their target genes. Most of the work for microRNAs is on the extraction of its regulation of genes [26,27]. Tool for the extraction of microRNA's association with biological processes and diseases (aberrant process) are limited. Thus, we focused on extracting connections between microRNAs and biological processes and diseases.

Although there are many ways by which microRNAs are connected with biological concepts such as other entities, biological processes, diseases, etc., we have found that the connections as stated in the text can be categorized into a small number of relations such as *regulation*, *involvement,* and *association* and we call them **CAIR** (**C**onnections through **A**ssociation, **I**nvolvement, and **R**egulation). Sentence (1) below is an example of a CAIR relation. Note, the categorization of the individual CAIR (Involvement, Regulation or Association) relations is loosely based on the lexical

trigger such as “involved in”, “plays a role in” etc. for Involvement, “regulates”, “promotes”, “inhibits” etc. for Regulation and “associated with”, “correlated with” etc. for Association.

Example 1: miR-522 regulates cell proliferation.

Additionally, it soon became evident that CAIR relations are not specific to microRNAs and are generalizable to other entities such as genes and proteins as well. Thus, we developed a RE system to extract CAIR relations to extract connections between biological entities (microRNAs, genes, proteins) and biological processes and diseases. We developed text-mining tools to extract the (1) role of microRNA in diseases (miRiaD) and (2) impact of phosphorylation of proteins, which are direct applications of CAIR relations. Several other tools based on CAIR relations were also developed by others in our research lab, which are discussed in Section 1.4.

### 1.2.2 Comparison relations

**Association** relations, one of our CAIR relations, can be inferred from comparison sentences. Biomedical researchers conduct experiments to validate their hypotheses and infer associations between biological concepts and entities, such as microRNA and disease or therapy and outcome. In such studies, researchers make observations under two different scenarios (e.g., disease sample vs. control sample). When the differences between the groups are statistically significant, associations can be inferred. Comparative studies are prevalent in nearly every field of biomedical/clinical research. Thus, the development of automated techniques to identify such statements would be highly useful. An example of such a comparison statement is shown in Example 2 below.

Example 2: The **expression of miR-21** was lower in **lung cancer tissues** compared with **normal tissues**.

We developed a RE system [25] based on our EDG-based relation extraction framework to identify comparison sentences and also extract the various components (compared aspect, compared entities/scenarios, and scale of the comparison). The developed system identifies explicit comparative structures at the sentence level, where all the components of the comparison are present in the sentence. To the best of our knowledge, ours is the only work that attempts to cover a wide range of comparisons, capture all comparison components, and does not impose any restrictions on the type of compared entities. Besides inferring Association relations, our comparison RE system forms the basis of the development of DEXTER [24], a text-mining tool to extract differential expression level information of genes and microRNAs in diseases.

## 1.3 Text-Mining Applications

Based on the EDG-based RE framework and CAIR relations, we have developed text-mining (TM) applications, which are discussed below.

### 1.3.1 miRiaD: microRNA-disease association

Our first application focuses on extracting the role of microRNAs in diseases. microRNAs are increasingly being appreciated as critical players in human diseases, and questions concerning the role of microRNAs arise in many areas of biomedical research. We have developed miRiaD (microRNAs in association with Disease) [23], a text-mining tool that automatically extracts connections between microRNAs and diseases from the literature. miRiaD attempts to extract the "myriad" ways the microRNA and disease are connected such as in Example 3 below. A unique feature of

our tool is that it not only detects the miRNA-disease connections but also detects any mention of the role of microRNAs in diseases. There are several possible roles that a microRNA can play in disease including its effect on the expression of its target genes, the outcome of a disease, or serve as a biomarker or therapeutic target for disease. Additionally, the microRNA may be involved in some cellular processes that are highly related to a disease, thereby explaining the microRNA-disease association.

Example 3: **MicroRNA-372** is associated with poor prognosis in **colorectal cancer**.

Using miRiaD, we developed a comprehensive microRNA-disease association resource/database and interface that can be used to answer various questions on microRNA's direct and indirect role in diseases.

### 1.3.2 Phos2X: Impact of protein phosphorylation

The second text-mining application focused on extracting the impact of phosphorylation on the substrate's (phosphorylated protein) function. This application demonstrates the generality and applicability of our EDG-based RE framework and CAIR relations.

Post-translational modifications (PTM) are chemical modifications of amino acid residues (site) of proteins (substrate) by a catalyst protein (enzyme). Phosphorylation is one of the most common forms of PTM. Often protein phosphorylation on different sites has functional implications on the substrate. For instance, proteins can be phosphorylated on different residues, leading to either alternative subcellular locations and/or interaction with distinct binding partners such as in Example 4. The functional impact of phosphorylation on the substrate properties

is not yet well represented in public databases. However, this information is critical for the understanding of protein networks and the prediction of functional outcomes.

Example 4: **Phosphorylation** of PTEN on Ser-380 impaired its **interaction** with Cdh1.

The work presented in this dissertation is concerned with the extraction of the impact of protein phosphorylation on (1) substrate's interaction with other proteins (binding partners), (2) alternative subcellular location of the substrate, (3) subsequent further post-translational modification (acetylation, ubiquitination, etc.) of the substrate. Also, to be able to detect the impacted protein properties, we have developed several protein-specific specific RE systems to extract protein-protein interaction, subcellular localization, and post-translation modification relations. All these were developed using our EDG relation extraction framework. We believe this work can be extended to impacts on other functions of the substrate protein as well as the impact of other post-translational modifications such as ubiquitination, glycosylation, etc.

### 1.3.3 DEXTER: Expression level information in disease

The next part of our work substantially applied the comparison RE system in developing a tool that detects the differential expression level information of microRNAs (and genes) in diseases from literature.

Gene expression is the process by which information from a gene is used in the synthesis of a functional gene product. The expression can have a profound effect on the functions of the gene and is the most fundamental level at which the genotype gives rise to the organism phenotype (observable traits). Additionally, abnormal gene expression disrupts biological processes and can lead to diseases. Identifying

expression profiles is useful for clinical research, diagnostics, and prognostics of diseases. There are several databases derived from high throughput transcriptomics data that capture gene/microRNA expression profiles. For example, BioXpress [28] is a gene expression and cancer association database in which expression levels are mapped to genes using RNA-seq data repositories such as The Cancer Genome Atlas [29], International Cancer Genome Consortium [30], Expression Atlas [31]. Linking such experimental data to existing knowledge from literature will be very useful for researchers.

The expression level information of a microRNA in disease tissue/cell/cell-line might not necessarily imply that the microRNA has a role in the disease but are important to capture since abnormal expression level can guide disease diagnosis, assess prognosis, or predict response to therapy. To demonstrate the generality and wide application of my work, we developed a tool called **DEXTER**, which extracts text evidence of genes/microRNA differential expression level in a diseased state (e.g. expression level of miR in cancer) compared to a normal/non-diseased state. While there are other databases, which capture such information, we go beyond by capturing contextual information, which may be important while interpreting expression profiles. A sentence containing differential expression information was shown in Example 2.

### 1.4 Thesis Contributions

Below we summarize the contributions of the different aspects of the dissertation.

- We developed a relation extraction framework based on EDG [3] that reduces the effort of developing individual relation extraction (RE) systems. We used this framework to develop different RE systems to extract CAIR relations. The initial

notion of CAIR relations was introduced and published in [32]. In addition to CAIR relation, the framework was used to extract comparison, protein-protein interaction (PPI), and subcellular localization relations.

The fast development of different RE systems is the major contribution of this framework. This is exemplified by the fact that in addition to the RE systems (CAIR, comparison relations, etc.) and TM tools (miRiaD, DEXTER, Impact of Phosphorylation), developed by me and described in this dissertation, several other RE systems were developed using this framework by others in our research lab. These include (1) eGARD [33], a tool to extract the impact of genomic anomalies on drug responses, glycosylation relation extraction, (2) various RE components in the text-mined miRNA resource, emiRIT [34], and (3) DiMeX [9], a mutation-to-disease association tool. Additionally, the thematic dependency representation motivated by EDG [3] and used in all our RE systems can enable machine learning applications to generalize more easily as shown in PPI extraction [3,4], chemical disease relation extraction [35]. We have started to include certain rule Open Information-Extraction (OpenIE) rules (active, passive, nominalized, relative clause, etc.) based on the tree families of Lexicalized Tree-Adjoining Grammars (LTAG) [36] for transitive verbs in our framework, which can be used in developing thematic representations for Machine Learning applications.

- We developed miRiaD [23], a tool to extract the role of microRNAs in diseases, which can be used to answer various questions on microRNA's direct and indirect role in diseases. miRiaD's results have been used by other people to create a miRNA resource called emiRIT [34]. emiRIT is an informatics portal with mined microRNAs in biological networks that incorporates text mined results from

miRiaD along with certain improvement (detection of biological processes) and additions (identification of extracellular miRs) using the EDG-based RE framework. miRiaD was applied on all MEDLINE abstracts till May 2020 and text-mined results were integrated into emiRIT. emiRIT contains information about 3,099 microRNAs, 255 diseases and 12,300 microRNA-disease associations from 121,371 abstracts.

- We developed a Phos2X, a tool to extract the functional impact of protein phosphorylation, which is critical for the understanding of protein networks and the prediction of the functional outcomes. Additionally, the impact of PTMs on processes and diseases is being recognized as evident from the increase in review articles such as epigenetic modification association to Alzheimer's disease [37], role of acetylation and methylation in Atherosclerosis [38], effect of phosphorylation, and ubiquitination on Parkinson's disease [39]. We believe that our impact tool can be adapted and extended to other post-translation modifications (PTMs) such as acetylation, methylation, etc., and other impacts such as biological processes and diseases. Some efforts by other researchers in our lab are underway in this respect.
- We developed a comprehensive RE system to automatically identify comparative structures from the text [25], which is essential for the development of our TM tool (DEXTER) to extract differential expression levels of genes and microRNAs in diseases. In addition, to be helpful for the development of DEXTER, our comparison RE system has wide applicability as comparative studies are prevalent in nearly every field of biomedical/clinical research such as comparing drug/treatment efficacy and outcomes in clinical trials. Biomedical researchers

conduct experiments to validate their hypotheses and infer associations and results of such experiments are stated in text through comparison statements.

- We have developed DEXTER [24], a tool to extract microRNA/gene expression level information in diseases. This expression information mined from literature can be used to extend transcriptomics-based expression databases. We processed the entire MEDLINE literature for differential expression information in cancer, and 24, 416 entries were made available for curation into the literature portion of BioXpress [28], an experimental expression database. Different works have been published as a result of this integration of DEXTER data with BioXpress such as the Identification of key differentially expressed MicroRNAs in cancer patients through pan-cancer analysis [40], and OncoMX: A Knowledgebase for Exploring Cancer Biomarkers in the Context of Related Cancer and Healthy Data [41].

## 1.5 Dissertation Outline

In chapter 2, we will introduce and motivate the CAIR relations, which will be important in connecting microRNAs to associated concepts (process, diseases). In this chapter, we also will describe the methodology behind extracting the discussed relations. Specifically, we will discuss how we use and extend the Extended Dependency Graph (EDG) representation to develop a relation extraction (RE) framework and extract CAIR relations. In chapter 3, we will discuss miRiaD, a microRNA-disease association extraction system. In Chapter 4, we will describe a tool to extract the functional impact of protein phosphorylation. This chapter demonstrates the generality and wide applicability of CAIR and the RE framework by going beyond microRNAs. In chapter 5, we will describe a tool to automatically identify comparative structures from the text, which is important for extracting association

relations and also essential in extracting differential expression information, which forms the basis of the next TM tool, DEXTER. In Chapter 6, we will describe a tool called DEXTER, which automatically extracts gene/microRNA differential expression level information in diseases. Finally, we conclude with a summary of the contributions and future work in Chapter 7.

# Chapter 2

# A RELATION EXTRACTION FRAMEWORK: APPLICATION FOR CAIR RELATIONS

## 2.1 Introduction

A major part of my work involves the development of CAIR (Connections through association, involvement, and regulation) relation extraction. To test the applicability of such relations, we have developed different text-mining (TM) tools, which rely on the extraction of CAIR relations. Some of these tools, as described in Chapters 4 and 5, require extraction of additional relations such as protein-protein interaction (PPI), post-translational modifications (PTM), and subcellular localization relations.

Since we are interested in the extraction of a wide range of relations such as CAIR and PPI relations, I was motivated in building a general framework that reduces the effort of developing individual relation extraction (RE) systems by handling certain common aspects of such RE systems, such as sentence splitting, parsing, syntactic dependency conversion, handling common linguistic constructs, and developing a common template for writing RE extraction rules. In this chapter, we present a relation extraction framework to facilitate the development of pattern-based RE systems.

The biggest benefit of this framework is to users with limited experience with Natural Language Processing, who can rapidly build various RE systems to extract relations of interest using this framework. Any user developing a new RE system will

simply have to state lexico-syntactic patterns at a level of generalization provided by syntactic dependencies. This fast development of different RE systems is the major contribution of the framework. This is exemplified by the fact that in addition to the RE systems (CAIR, comparison relations, etc.) and TM tools (miRiaD [23], DEXTER [24], Impact of Phosphorylation), developed as part of this dissertation work, several other RE systems have been developed using this framework by others in our research lab. These include eGARD [33], a tool to extract the impact of genomic anomalies on drug responses, various RE components in the text-mined miRNA resource, emiRIT [34], and DiMeX [9], a mutation-to-disease association tool. In addition, RE systems are being developed using this framework for glycosylation relation extraction and to connect variants with Alzheimer's disease.

The relation extraction framework is based on extracting relations patterned on syntactic dependencies, which are grammatical relations between words in a sentence. Additionally, our framework utilizes notions of (1) adding thematic dependencies (or numbered-arguments) for extracting arguments of relation and (2) identifying extra-syntactic (referential) information between a syntactic argument and the actual target argument, which were proposed in an earlier work: Extended Dependency Graph (EDG) [10]. EDG is a new representation to account for and unify the various syntactic variations (active, passive, nominalized, etc.) and incorporate extra-syntactic information (is-a relations) and move towards more thematic dependencies. Additionally, we handle cases of elliptical constructions, which involve sentence constructions, where an argument for a predicate trigger is omitted, but implied (e.g., "**MiR-31** inhibits cell **migration** by targeting **SATB2"**). Here, the subject of "targeting",

which is “mir-31” is missing/elided based on syntactic dependencies. We call these cases null-arguments.

Before we describe our RE framework, we will motivate the use of syntactic dependencies in Section 2.2.1 and introduce the concept of EDG and the numbered-argument representation and the motivation behind it in Section 2.2.2. In Section 2.3, we describe the architecture of RE framework and describe the details of the RE framework based on the notions of EDG. Recall one of the motivations behind developing the RE framework was the extraction of CAIR relations. Following the description of the framework, we will motivate CAIR relations in Section 2.4 and show how it can extract through the RE framework in Section 2.5.

## 2.2 Background

### 2.2.1 Approach to Relation Extraction: Using Syntactic Dependencies

In our approach to relation extraction, we are focused on the extraction of the relations that are stated in the text through lexical triggers (predicate of a relation). For example, “Regulation” relation between an entity (gene or microRNA) and biological process are conveyed through different words/phrases such as “regulates”, “promotes”, and “inhibits”.

We use a common approach to extracting predicate-argument relations that is based on syntactic dependencies. Note, due to the lack of dependency parsers trained on biomedical text, we first use constituency parser specifically trained on biomedical text to obtain a parse tree and then convert the parse tree to syntactic dependencies using a converter. We use the Charniak-Johnson parser [42,43] with David McClosky’s adaptation to the biomedical domain [44] to obtain constituency parse

trees for each sentence. We convert the syntactic parse tree into syntactic dependencies using the Stanford conversion tool [45,46] . In Figures 2.1 and 2.2, the Constituency parse tree and the Standard Dependency Graph (SDG) [46] using Universal Dependency notation [47] for the sentence “miR-21 regulates cell proliferation” are shown, respectively. We use the syntactic dependency graph representation to define patterns for relation extraction since it provides a representation closer to the predicate-argument relations than parse trees.

The syntactic dependency graph provides a representation of grammatical relations between words in a sentence. One such dependency triplet as shown in Figure 2.2 is nsubj(miR-21, regulates), where the relation is nsubj (nominal subject), the governor and dependent of the relation being “regulates” and “miR-21” respectively. In this case, the Regulation relation between miR-9” and the biological process “cell proliferation” correspond to the two syntactic dependents of the lexical trigger “regulates”.

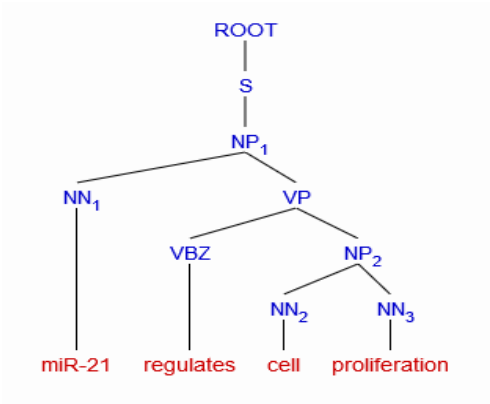


Figure 2.1: Example Constituency Parse Tree

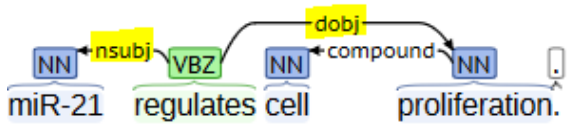


Figure 2.2: Example Stanford Dependency Graph (SDG)

Syntactic dependencies allows us to examine sentences at an abstraction level that abstracts away from many textual variations that are not important for extracting predicate-argument relations. For example, the same dependencies for Regulation are obtained from "miR-21 regulates proliferation" as well as "miR-21 may regulate proliferation". Of course, the use of the modal "may" in the latter sentence does provide information that may be critical for other purposes. However, the use of the dependency structure provides a uniform representation as far as the connection between "miR-21" and "proliferation" is concerned.

Text in biomedical literature is often more complex and dense with information. While syntactic parsing provides an ability to abstract away from some textual variations, some simple syntactic variations provide different dependency structures, as can be seen from the dependencies (edge labels above the sentence) in Figures 2.3-2.5. To abstract away from such syntactic variations, EDG proposed dependencies (numbered argument edges: Arg_0/1) that are more thematic in nature. In the next subsection, we will introduce the EDG representation and motivate the reason behind our adaption of numbered-argument representation in our framework.

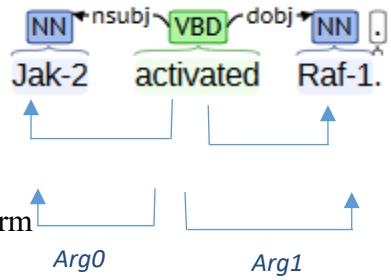


Figure 2.3: Active Verb Form

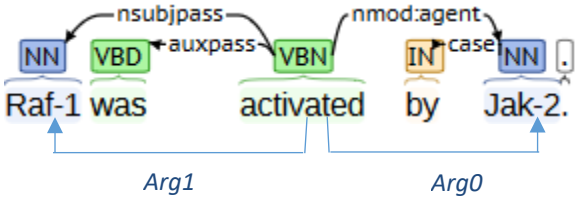


Figure 2.4: Passive Verb Form

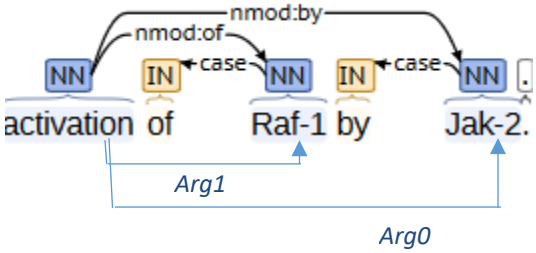


Figure 2.5: Nominalized Verb Form

### 2.2.2 Extended Dependency Graph

The syntactic variations depicted above are systematic and have been captured in various linguistic theories as well as grammatical frameworks (e.g., tree families of Lexicalized Tree-Adjoining Grammars (LTAG) [36]. The use of thematic dependencies can allow us to abstract away from the variations in the syntactic dependencies. However, there are a large number of thematic dependencies and subtle

differences between them (e.g., between recipient, beneficiary, and theme) are not immediately relevant for our purposes. Since nearly all our relations are binary (between two arguments of a trigger or lexical anchor), we assume that calling them Arg_0 and Arg_1 will suffice. This again is similar to LTAG use of NP_0 and NP_1 nodes as well as to a significant reduction on thematic roles in Propbank [48]. In our approach, the same (thematic) predicate-argument relations of Arg_0(activate, Jak_2) and Arg_1(activate, Raf-1) will be produced corresponding to the text in Figures 2.3-2.5. To provide for such generalized representation and to account for other generalizations, we had proposed **Extended Dependency Graph (EDG)** in [3] to include information about the text that goes beyond syntax motivated by tree-families of LTAG, where the same thematic relation (two arguments with a lexical anchor in our case) have different syntactic structures.

Our discussion above showed how a unified representation can be obtained for sentences in Figures 2.3-2.5 by considering Arg_0 and Arg_1 (henceforth called *numbered argument*) dependencies. Following these edges can provide the arguments for the relations between the proteins. However, the Arg_0 and Arg_1 edges also provide direct access to the arguments in the sentences in Figures 2.6-2.9. The first two correspond to the use of relative and reduced relative clauses (Figures 2.6 and 2.7).

EDG not only considers syntactic dependencies between words in a sentence but also utilizes information beyond syntax to capture different dependencies. Figures 8-9 show the use of extra-syntactic processing that allows new Arg_0 and Arg_1 edges that are not just based on the syntax. In Figure 2.8, the syntactic processing not only establishes an Arg_0 edge between “targets” and “oncogene” but also an *appos*

(appositive) link from the latter to HOX-11. The appositive relation corresponds to an Is-a relation and this Is-a information can be propagated to obtain an Arg_0 edge between “targets” and “HOX-11” on non-syntactic grounds. Figure 2.9 represents a similar case except that the Is-a information comes from “a member collection” relation (human Cdc2, Cdk2, and Cdk3 are members of a collection of cyclin-dependent kinases that interact with Cdi1) and similarly propagated.

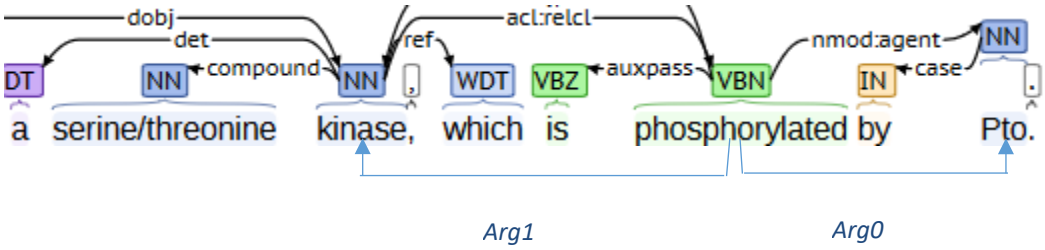


Figure 2.6: Relative Clause

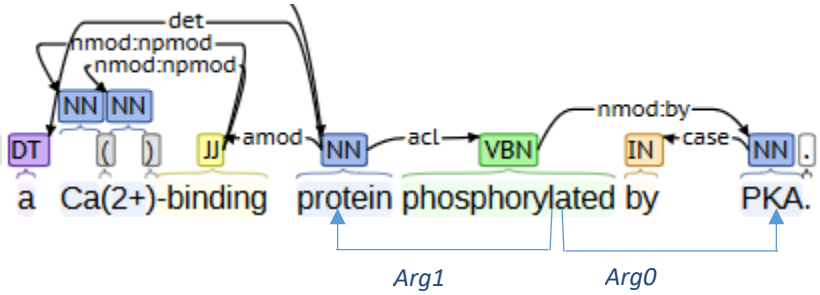


Figure 2.7: Reduced Relative Clause

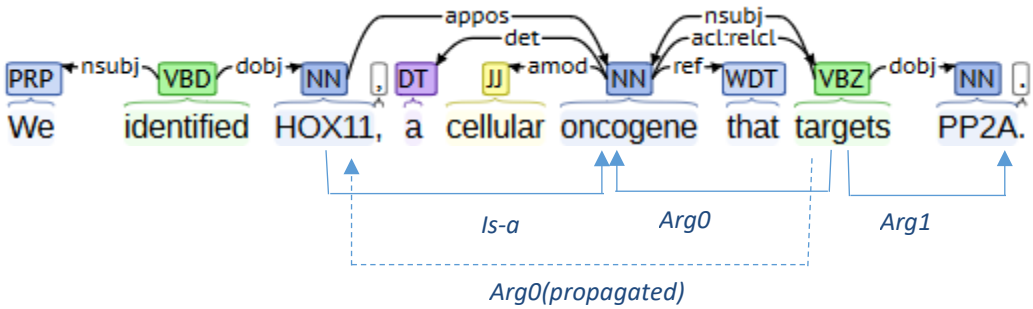


Figure 2.8: Propagation of Is-a relation

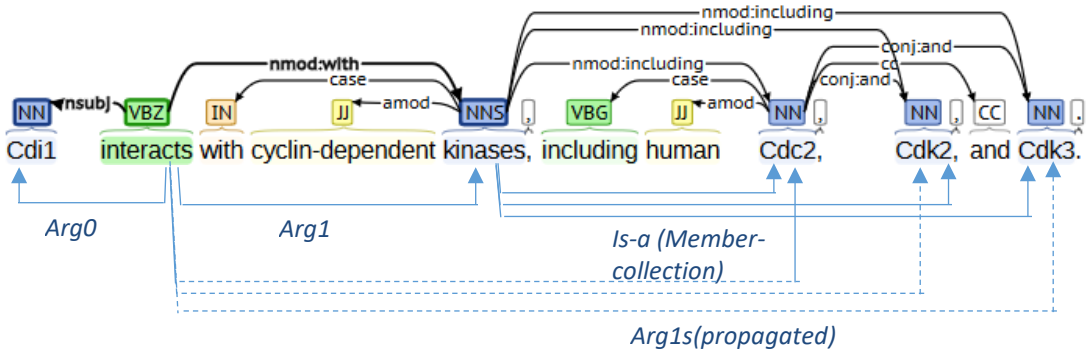


Figure 2.9: Propagation of Member-collection

We use the same idea of adding numbered arguments based on syntactic information in our relation extraction framework to extract relations. The primary reasons for using the notion of numbered arguments in our RE framework and the extraction of different relations (CAIR, PPI, etc.) are:

1. Further processing such entity typing of the arguments of an extracted relation is easier since a TM tool only has to check the type of Noun-Phrases headed by the Arg0 and Arg1. For example, is Figures 2.3-2.9 are all examples of Protein-Protein interaction relations, where the Arg0 and Arg1 edges point to phrases, which are proteins.

2. Numbered argument representation provides "consistent labels across different syntactic realization of the same verb [49]" making generalization easier downstream. For example, this representation can enable machine learning applications to generalize more easily as was shown in [3,4,35].

3. The final motivation behind using Arg0/Arg1 edges is the propagation of these numbered argument edges (and not any syntactic edge) using reasoning that goes beyond syntax as shown in Figures 2.8 and 2.9.

Having motivated the reasoning behind developing RE systems, which incorporate the numbered-argument representation based on syntactic dependencies, we will describe the details of our Relation Extraction Framework in the next section.

## 2.3 EDG-based RE Framework

We developed an RE framework based on the notions of EDG (numbered arguments, extra-syntactic processing) for not only extracting the relations of interest in this work but also enables others to extract other relations with little effort. There are several limitations of our previous work on [3]. It was mostly an idea of adding numbered argument edges and some extra-syntactic information and this idea was exemplified by describing some rules for protein-protein interaction (PPI) relations and a general method for considering all aspects of biomedical relations extraction had not been considered.

In developing the EDG framework, we have included functionalities for many aspects such as sentence splitting/tokenization, parsing, syntactic dependency conversion, which are common to different RE systems. The inclusion of these functionalities enables us to develop new RE systems for different relations by focusing only on the lexical triggers for the relations and the mapping of syntactic

dependencies to numbered arguments. Additionally, this framework also accounts for syntactic variations by incorporating the various EDG generalizations such as the notion of numbered-arguments, adding extra-syntactic information (is-a), handling elliptical constructions (null arguments), and propagation of arguments as discussed. The framework also provides a single template for writing rules for relation extraction based on patterns involving word lemma, part-of-speech tags (POS), and dependency edge labels which are then matched and applied using Stanford Semgrex [50].

### 2.3.1 Architecture of the Framework

The different steps of our framework are depicted in Figure 2.10, which includes: sentence splitting and tokenization, constituency parsing, dependency conversion, and pattern matching based on some set of lexico-syntactic rules. In this framework, we have allowed for different rule phases, where a different set of rules can be provided as input in each phase. This division into phases allows separating rules common to any RE task from RE-specific rules. We expect that the extraction of any relation should involve the modules which are shaded in Figure 2.10. Since these are common to all relation extraction, we have developed them and included them as part of the framework. Thus, the designer only has to write the part of rules (called Phase2 in Figure 2.10) that correspond to their relation of interest. We will discuss the different sets of rules and their organization in different phases in the next subsection 2.4.2. Note, certain rule templates (active, passive, nominalized, relative clause, etc.) based on the tree families of Lexicalized Tree-Adjoining Grammars (LTAG) [36] for transitive verb has been included in the framework (as Phase 2 set of rules) as a reference for users developing their own Phase 2 RE-specific rules.

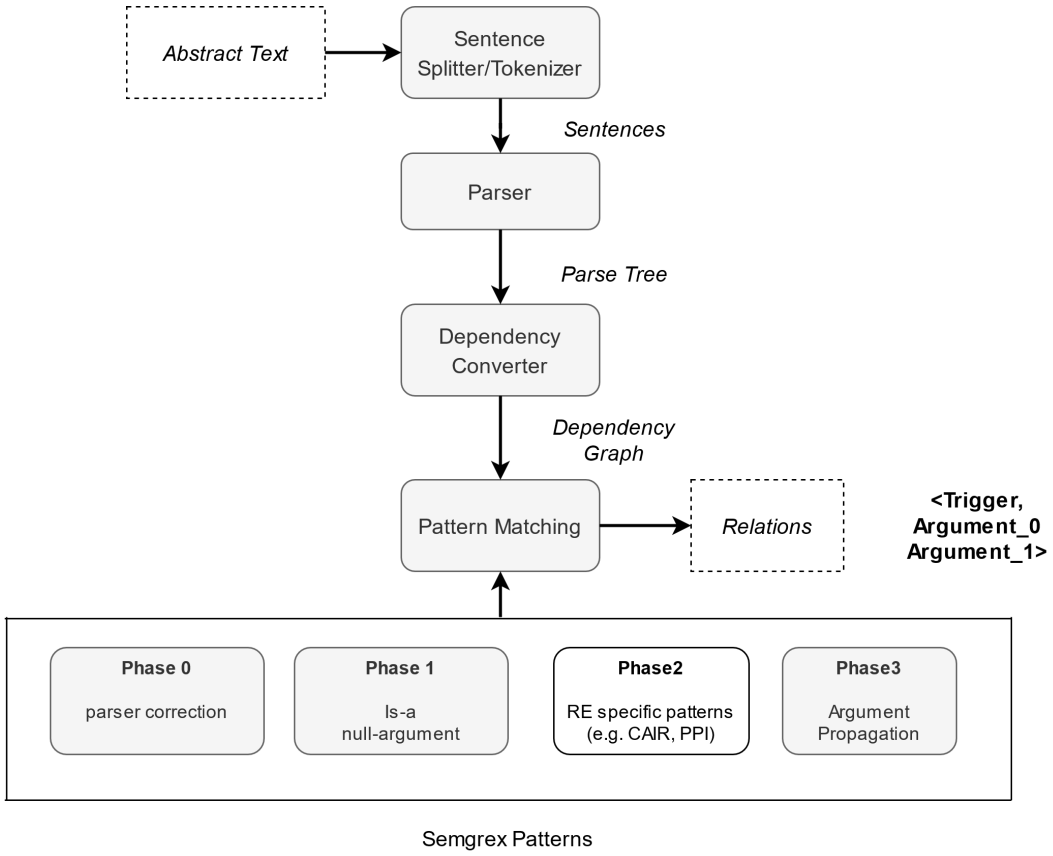


Figure 2.10: Relation Extraction Framework

Below we will describe the systems that we use in the different steps of the framework. Given an input text, typically a Medline abstract, we first tokenize and split the text into sentences using the Stanford CoreNLP toolkit [45]. We then use the Charniak-Johnson parser [42,43] with David McClosky's adaptation to the biomedical domain [44] to obtain constituency parse trees for each sentence. Next, we use the Stanford conversion tool [45,46] to convert the parse tree into syntactic dependencies. In Figures 2.2a and 2.2b, the constituency parse tree and the syntactic dependencies

after applying the conversion tool for the sentence "*miR-21 regulates cell proliferation*" were shown, respectively.

We use the "CCProcessed" representation (an option in the Stanford conversion tool), which collapses and propagates dependencies allowing for appropriate treatment of sentences that involve conjunctions. Consider the syntactic dependency graph for the sentence "*mir-21 regulates proliferation and migration*" as shown in Figure 2.11 below. In this example, the mir-21 regulates two processes ("proliferation" and "migration"), which are connected through a conjunction. The "CCProcessed" option propagates the *dobj* from the trigger "regulates" to the head to the second conjuct ("migration"). This propagated *dobj* edge would be absent if the "CCprocessed" option was not provided to the conversion tool. The "CCProccessed" option also propagates dependencies involving prepositions, conjuncts, as well as the referent of relative clauses are "collapsed" to get direct dependencies between context words.

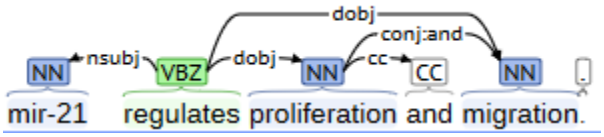


Figure 2.11: Example Dependency Graph with "CCprocessed" option

Based on this syntactic dependencies representation (Figure 2.11), patterns (Semgrex patterns [50]) are defined to add additional edges (is-a, numbered argument, etc.). For example, a lexico-syntactic pattern ***{lemma:regulate} > nsubj*** and ***{lemma:regulate} > dobj,*** indicating to follow the *nsubj* and *dobj* syntactic edges

from a token with lemma “regulate” can be used to add Arg0 and Arg1 edges for a regulation relation. For dependency graph shown in Figure 2.11, the above pattern will add an Arg0 edge to “mir-21” and two Arg1 edges to “proliferation” and “migration” from the trigger “regulate”.

We have developed a standard template to write these lexico-syntactic rules to add new edges (numbered or otherwise). The rules are based on specifying a set of conditions and actions. The set of conditions are based on Stanford Semgrex [50], which can be used to specify patterns based on the lemma, part-of-speech tags (POS), and dependency edge labels. The developer can specify associated actions to add the new edges (numbered or otherwise) if the conditions are met. Details of the standard template to write the rules (set of Semgrex conditions and associated actions), and examples can be found in Appendix A. An illustration of the working of the RE framework with a running example can be found in Appendix E.

### 2.3.2 Different Types of Rules

There are several types of rules that we expect to use in this EDG framework in the extraction of a relation. Some of these are independent of the specific relation to be extracted and some are developed specifically for the relation of interest. These task-independent set of rules add edges at the dependency level such that a user can focus on just writing their own RE specific rules for numbered arguments addition without concerning themselves with syntactic variations (e.g., relative clause, null-arguments) or even argument propagation based on extra-syntactic information such as is-a relations. This task-independent set of rules include patterns to handle:

a) Parsing Errors
b) Extra-syntactic information (is-a relations)

c) Null-argument or elliptical constructions

d) Numbered-Argument Propagation

Rules to handle **(a) Parsing Errors** correct some common mistakes occasionally made by the parser. These were developed based on our observations of the particular parser (BLLIP) used in the framework. Since this set of rules are specific to the parser and the version used, we discuss the specific set of rules for handling BLLIP parsing errors in Appendix B.

The set of rules for handling **(b) Extra-syntactic information** that captures information beyond syntax as exemplified in Example 1 below (dependency structure in Figure 2.8 shown earlier). These set of rules add referential edges to the syntactic dependency graph, which will be used in propagating argument edges in later phases. In Example 1, a RE specific rule for protein-protein interaction (PPI) that doesn't consider the appositive construct between "HOX11" and "oncogene", will only identify "oncogene" as an argument of the trigger "targets". We need to (1) identify such extra-syntactic structures (here is-a relation indicated through appositive) and (2) propagate the numbered-argument edge added by RE specific rule to the appropriate argument ("HOX11" in this case) using the extra-syntactic information. Thus, rules for handling (b) Extra-syntactic information include capturing is-a relations. These will be described in detail in Section 2.4.2.1. Rules for **(d) Numbered-Argument Propagation** contain patterns for propagating the numbered-argument edge added by RE specific rules, which will be described in detail in Section 2.4.2.2.

Example 1: We identified **HOX11**, a cellular **oncogene** that <u>targets</u> <u>**PP2A**</u>.

Set of rules for handling **(c) Null-argument or elliptical constructions** involves sentence constructions, where an argument for a predicate trigger is missing or elided based on syntactic dependencies as in Example 2a, below. When the argument is omitted, but implied, we call them **null-arguments**. Example 2b, depicts how the sentence could have been rephrased if the argument was not elided. Note, this elided argument information is present and can be obtained through parse trees by using the null pronoun PRO tag, which is a pronominal determiner phrase (DP) that is postulated in the subject positions of non-finite clauses [51]. But since we are converting the parse tree to syntactic dependencies, this elided argument information through PRO is lost. In these cases, there is no direct dependency between "miR-31" and the trigger "targeting". We have developed rules patterned on syntactic dependencies to recover this elided information from such cases and are included in this set (c). This set of rules will be described in Section 2.4.2.3.

Example 2a: **MiR-31** <u>inhibits</u> cell **migration** by <u>targeting</u> **SATB2**.

Example 2b: **MiR-31** <u>inhibits</u> cell **migration** by [MiR-31 $_{\text{PRO/NULL}}$]<u>targeting</u> **SATB2**.

We provide all these task-independent rules as part of the framework, but users can modify them based on specific requirements (e.g., change in the parser version). These task-independent rules along with the user-defined relation specific rules are applied by the framework to extract the relation of interest. Note, certain sets of rules need to be applied before others could be applied i.e., the order of the rule phases is important. Rules to handle (a) Parsing errors need to be applied before any task-independent or user-specific rules are applied. Additionally, (b) Extra-syntactic

information and (c) Null-arguments rules need to be applied before any RE specific rules for adding numbered-argument edges are applied. Also, since (d) Propagation rules use edges added in task-independent rules and numbered-argument edges, it should be applied at the very end. Thus, we provide an ordering of rules in our framework (also indicated in Figure 2.10) as follows:

Phase 0: (a) Parsing Errors

Phase 1: (b) Extra-syntactic information (is-a edges), and (c) Null-argument

Phase 2: Relation specific rules (CAIR, PPI) for adding argument edges

Phase 3: Numbered Argument Propagation

#### 2.3.2.1 Extra-syntactic Information: Is-a relations

As indicated before capturing certain extra-syntactic information would be helpful to propagate the numbered-argument edges to the correct tokens. Below we will discuss rules to add such extra-syntactic information: is-a relations

***Is-a:*** In this set of rules, we add edges to capture is-a relation, which captures if an entity has a certain **property**. Consider the example in Figure 2.12a below, where "HOX-11" has the property of being an oncogene indicated by the appositive structure ("appos" edge). Capturing such is-a relation is important for our framework since a RE-specific rule in Phase 2, which does not consider this "appos" link will not be able to extract "HOX-11" as one of the arguments of the predicate "target".

One of the motivations behind the development of this RE framework was the fast development of different RE systems. Researchers who are not familiar with NLP do not need to worry about accounting for such grammatical constructs. Hence, to reduce the burden of writing a huge number of relation-specific rules in Phase 2, we handle certain constructs such as "is-a" relations, which are independent of any RE

task. In this example first, we add an “is-a” edge based on the *appos* edge (appositive) between “HOX-11” and “oncogene”. A RE-specific rule only based on *nsubj* and *dobj* edges will add numbered argument edges Arg0 and Arg1 to “oncogene” and “PP2A” respectively. We then propagate the Arg0 edge based on this newly added “is-a” edge to “HOX-11” and thereby extract it as the argument of “targets”. Note the Numbered Argument propagation rules are the last set of rules (included in Phase 3) and are discussed in the next subsection.

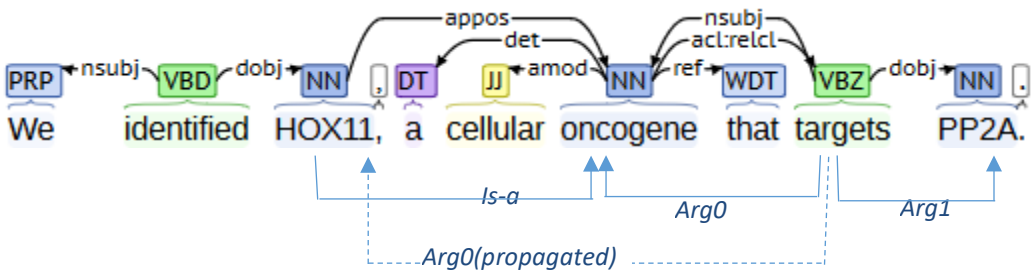


Figure 2.12a: Propagation of Is-a relation

In addition to certain grammatical constructs like appositives and copular, which indicate Is_a relation, we use certain phrases to detect Is-a relation such as acts as, functions as, serves as, etc. Consider the copular case in Figure 2.12b In these cases, we use the *cop* edge and *nsubj* edge to add the is_a edge. The example sentence in Figure 2.12c contains an Is-a relation through the phrase “serves as”. In these cases, we use the *nsubj* edge and *nmod:as* from the trigger to add the *is-a* edge. Additionally, in both these cases, we also check for the presence of the *det* edge to the articles “a” or “an” (Only applicable for singular verb forms). These is-a relations are also used

directly in our miRiaD tool to extract biomarker and therapeutic target relations (discussed in Chapter 3, Section 3.3.1.2), which are basically properties of microRNAs.

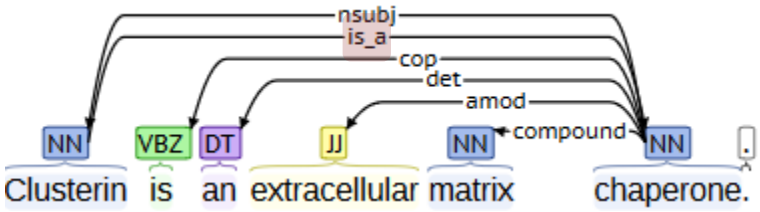


Figure 2.12b: Example 1 EDG for Is-a

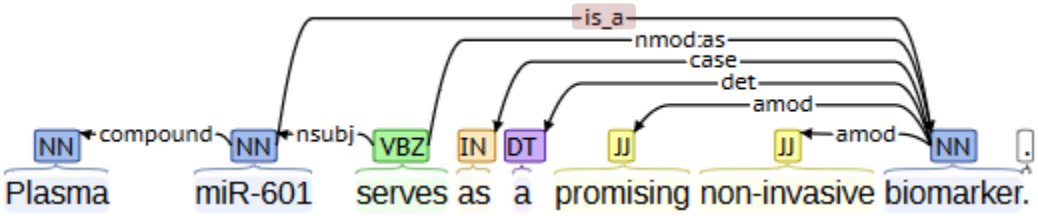


Figure 2.12c: Example 2 EDG for Is-a

**Another category** of structures, which convey **Is-a** information involves cases, where an entity is part of a set/larger collection (e.g., Y, including X1, X2, X3). We name these constructs **member-collection**. These relations are used to link a generic reference (called collection) to a group of entity mentions (called members). Like in Figure 2.9 (Page 24), typical keywords that can identify is-a relation through member-collection construction are “including” and “such as”. We use *nmod:including* or *nmod:suchas* dependency edges to add “member-collection” edges. Note, even though we call these edges “member-collection”, these are semantically

equivalent to is-a relations as they indicate the member entity belonging to the collection entity and thus can be used to propagate other edges (numbered or otherwise).

#### 2.3.2.2 Propagation

In Figure 2.12a, rules for adding numbered argument edges using only syntactic processing only establishes an Arg_0 edge between "targets" and "oncogene". But in Phase 1, we will add the extra-syntactic edge "is-a" using the *appos* (appositive) link to HOX-11. This Is-a information can be propagated to obtain an Arg_0 edge between "targets" and "HOX-11" on non-syntactic grounds. Figure 2.9 shown earlier represents a similar case except that the Is-a information comes from "a member collection" relation (human Cdc2, Cdk2, and Cdk3 are members of a collection of cyclin-dependent kinases that interact with Cdi1) and similarly propagated.

#### 2.3.2.3 Null argument

Connections between genes/microRNA and complex processes or disease-related concepts are multi-step and are typically stated through CAIR relations. Sometimes authors may use certain structures to further explain these connections, where one of the arguments of a predicate trigger is elided as discussed earlier.

Consider the sentence in Figure 2.13. In this example, "miR-126" is clearly the Arg_0 argument for both the "inhibit" and "suppressing" predicate trigger and this can be inferred based on the syntactic dependencies. However, there is no direct syntactic dependency between the VBG node for "suppressing" and "miR-126". "miR-126" should be considered an Arg_0 argument for this verb ("suppressing") as well. The

agent argument of "suppressing" is implicit because stylistically it would be awkward to include it a second time or even by including a pronoun. Because it is awkward to mention the agent each time one of its predicates is used, the agent is mentioned only once, in the beginning, and omitted in the second case (i.e., for the verb suppress). We call such cases in which an argument is not explicitly stated but inferred from an earlier mention as a "null-argument" of a predicate. We borrow this terminology from the use of NULL nodes in various linguistic theories [51]. The ideas for null argument rules that identify the agent for the second predicate were taken from RLIMS-P [52,53] and iXtractR [54]. We present below one example of handling null argument structures. Rules for handling other null argument structures can be found in Appendix C.

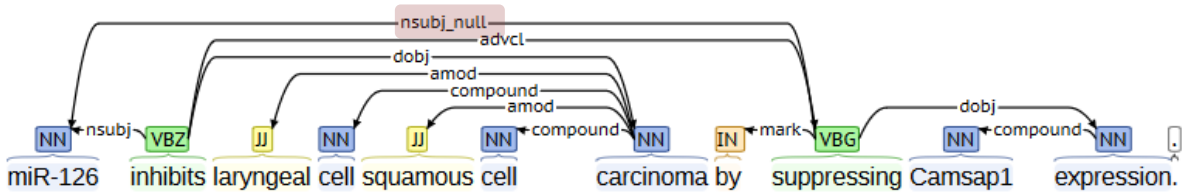


Figure 2.13: Null Argument Handling Example

The null-argument handling starts with the addition of an edge *nsubj_null* which is later changed into Arg_0. The addition of the *nsubj_null* is handled not by a specific trigger but by the presence of a VBG node which has an outgoing *mark* edge to the preposition "by" and an incoming *advcl* edge from a verb. But as it would be incorrect to call the syntactic dependency *nsubj*, so instead we add *nsubj_null*

indicating it has been added through null argument structures. This addition of a **nsubj-like** edge enables our tool to minimize patterns/rules that will be required for adding numbered-argument edges (Arg_0 in this case).

Other instances of null arguments are triggered in clauses separated by prepositions "to" and an untensed verb, "via" or "through" with a nominal form of a verb. For example, "*Tumor suppressive miR-1 induces apoptosis through direct inhibition of SRSF9 in bladder cancer*" is an example of the latter case.

## 2.4 Motivating CAIR Relations

The motivation behind developing CAIR relations was to extract connections between genes/microRNAs and biological concepts such as processes and diseases. Although there are many ways by which biological entities (genes, proteins, and microRNAs) are connected with biological concepts such as other entities, biological processes, diseases, etc., we have found that the connections as stated in the text can be categorized into a small number of relations such as *regulation*, *involvement,* and *association* and we call them **CAIR** (**C**onnections through **A**ssociation, **I**nvolvement, and **R**egulation). Note, the categorization of the individual CAIR (Involvement, Regulation or Association) relations is loosely based on the lexical trigger such as "involved in", "plays a role in" etc. for Involvement, "regulates", "promotes", "inhibits" etc. for Regulation and "associated with", "correlated with" etc. for Association. Below we will introduce and motivate each CAIR relation and provide examples.

Consider biological concepts such as **biological processes** or **diseases**/disease-related concepts. Genes/microRNAs can play a significant part in affecting the biological processes and diseases and thus authors will typically talk about these

connections in terms of **Involvement**. A cellular/disease process typically involves hundreds of thousands of individual "micro" steps involving biological entities, such as regulation, binding interactions, post-translational modifications. An entity may be involved in a single such micro-step and yet it may be a significant step in the cellular/disease process. Therefore, authors may want to highlight the connection between this entity and the process. But for sake of succinctness, the connection may simply be stated as "X is involved in process A". Thus to capture the mentions of such connections (between genes/microRNAs and multi-step processes), we consider it important to extract such **Involvement** relations.

Authors can also state such involvement relations using other phrases such as "plays a role in", "implicated in", etc. (see Examples 3a, 3b below) [1]. Even though the relation between the gene/microRNA and the process is stated differently at the textual level, we will treat them as falling into the same relation of Involvement.

Example 3a: miR-224 <u>plays a role in</u> cell proliferation. [PMID: 22989374]

Example 3b: miRNA330-5p <u>has been implicated in</u> the progression of prostate, neuronal and pancreatic cancers. [PMID: 27633518]

Sometimes the connection between an entity and a process can be slightly more specific as exemplified by Examples 4a and 4b. In these sentences, the stated connections are more specific as it conveys that the entity regulates (i.e., control or maintain the rate or extent) of a process. These set of triggers convey more specific

---

[1] Individual words and multi-word expressions such as "plays a role in" and "implicated in" will be called (lexical) triggers for the Involvement relation. The complete list of triggers we use for Involvement as well as all other relations of interest are provided in Appendix C.

information than Involvement trigger and we distinguish it from Involvement relations. Thus, we introduce **Regulation** relations, a subtype of Involvement that is conveyed and extracted through lexical triggers such as “regulate”, “inhibit”, “promote”, “induces”, “mediates” etc.

Example 4a: miR-522 regulates cell proliferation. [PMID: 25131211]

Example 4b: MicroRNA-21 promotes cell growth and migration. [PMID: 25400316]

In texts that describe the findings or observations of biomedical experiments, it is common to note statements of correlations. Such correlations can be used to infer any causality or agentive role of one of the participants. The word “correlation” has a more technical meaning and hence we call the general relation as **Association** (see Examples 5a-5d below). Also, unlike Involvement, Association relation is usually symmetric.

Example 5a: microRNA-21 expression is associated with overall survival in patients with glioma.

Example 5b: The expression level of miR-409-3p was negatively correlated with osteosarcoma metastasis. [PMID: 26992637].

Example 5c: microRNA-132(miR-132) is linked with synaptic plasticity and cognitive impairment. [PMID: 26806865]

Example 5d: Cancer stem cells (CSCs) are linked to metastasis. [PMID: 27113763]

Notice that in both Examples 5a, and 5b, the relation connects the microRNA expression rather than microRNA itself.

## 2.5 Extracting CAIR Relations Using EDG-based Framework

A major focus of my dissertation became the development of a RE system to extract CAIR relations from text. CAIR relations form the basis of the development of the different text-mining applications described in this dissertation: (miRiaD [23], DEXTER [24], Impact of Phosphorylation). We have so far motivated the importance of CAIR relations and described the development of an EDG-based relation extraction framework. In this section, we will discuss how we use it for the extraction of the CAIR relations of Involvement, Regulation, and Association using the framework. This set of rules for extracting CAIR relations are RE-specific and hence are part of the Phase 3 set of rules of the framework. Note, the extraction of CAIR relations has not been directly evaluated but indirectly evaluated through the different text-mining (TM) tools that are based on CAIR discussed in Chapters 3 and 4. miRiaD [23] described in Chapter 3 and the tool to extract the Impact of Phosphorylation described in Chapter 4 are direct applications of CAIR relations and are evaluated. Additionally, another tool DiMeX [9], developed by another researcher uses CAIR relations and has been evaluated and published.

### 2.5.1 Involvement

Involvement relations involve predicate-argument relations, where generally the gene or microRNA will be the first argument and the connected concept such as biological process the second. There are two types of lexical triggers we need to consider to extract Involvement relations.

The first set of triggers are verb-based triggers that require prepositional phrases (PP) complements. Such PP complement verb phrases include “involved in”, “implicated in” etc. (A full list of triggers can be found in Appendix C.1). We

manually enumerate all the verb subcategorization for such triggers rather than using a resource such FrameNet [55] due to the relatively small number of involvement verb triggers. Consider the example in Figure 2.14, where we use the edges *nsubj* and *nmod:in* from the trigger "implicated" to add *arg0* and *arg1* edges respectively. Verb triggers with different prepositions will have a different syntactic edges for addition for *arg1*. For example, for the triggers such as "required for" we will use the *nmod:for* edge. A nominalized verb form can also be used in these cases (e.g. *Implication of miR-126 in the process of inflammation*). In these cases, we use the edge *nmod:of* from the nominalized trigger (e.g. "implication") to add *arg0*.

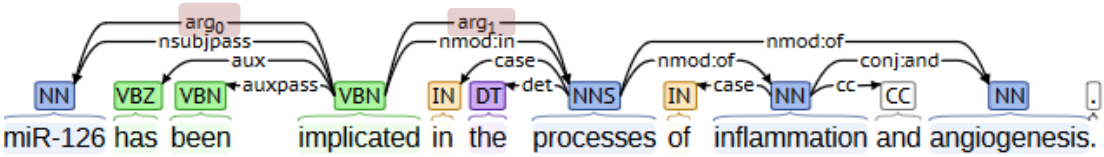


Figure 2.14: Example 1 EDG for Involvement

A second class of phrases, which indicates Involvement relation are multi-word triggers such as "plays/has a role in", "has an effect on". In these cases, the presence of the nouns (role or effect) rather than the verb (has or play) indicates the relations and thus needs to be considered in EDG rules. Consider the example sentence in Figure 2.15. Here we use the *nsubj* edge and *nmod:in* edge from "plays" to determine the *arg0* and *arg1* edges respectively. Additionally, we also need to consider the *dobj* edge from "plays" to "role" to add the arg edges. A full list of lexical triggers can be found in Appendix C.1 and syntactic variations can be found in Appendix D.

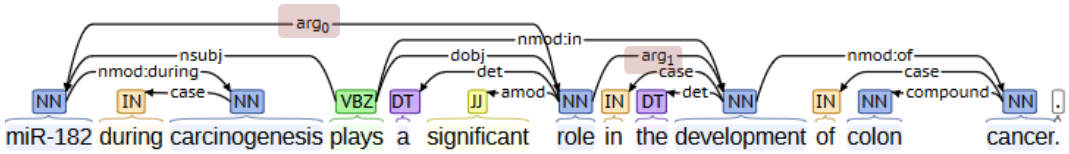


Figure 2.15: Example 2 EDG for Involvement

### 2.5.2 Regulation

Regulation relations are similar to involvement relations except for the list of trigger words/phrases that can serve in the predicate position. These predicate triggers are verb-based and have Noun Phrases as complements (rather than prepositional complements) such as “regulates”, “mediates”, “promotes”, “inhibits” etc. Here we used verb-based rules (active, passive, and normalized) to determine the arguments. An example of a Regulation relation from a sentence in the active form is presented in Figure 2.16. In this sentence, we follow the *nsubj* and *dobj* edges from the trigger “regulates” to add the arg0 and arg1 edges respectively. Notice the example sentence in Figure 2.17. This is a case of null-argument and as discussed in section 2.3.2.3, we add the edge *nsubj_null* in phase 2, from “regulating” to microRNA-520g, and hence we will the *nsubj_null* edge to add the *arg0* edge in this case.

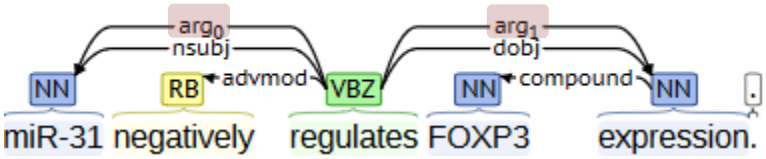


Figure 2.16: Example 1 EDG for Regulation

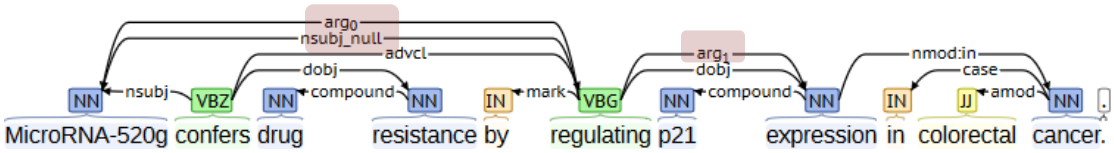


Figure 2.17: Example 2 EDG for Regulation

### 2.5.3 Association

Association relations are detected from predicate-argument relations, where the gene/microRNA or the state of the GPM will be one of the arguments of the relation and can be either first or second argument. Association trigger includes phrases, such as “associated with”, “correlated with/to”, “linked to” etc. The rules for adding arg edges for Association are very similar to Involvement trigger (verbs with PP complements). An example sentence is shown below in Figure 2.18. Here we follow *nusbjpass* edge to add *arg0* and nmod:with or *nmod:to* to add the *arg1* edge. As with any verb-based trigger, Association relations can also have nominalized cases. A full list of lexical triggers can be found in Appendix C.1 and syntactic variations can be found in Appendix D.

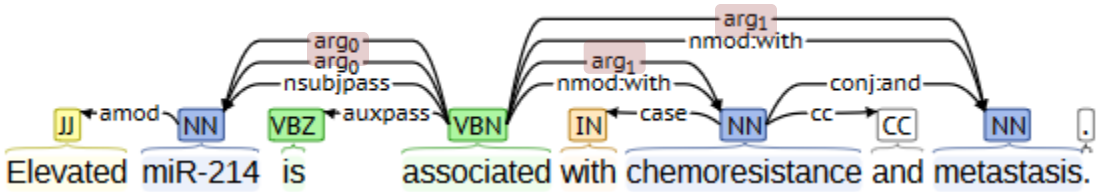


Figure 2.18: Example EDG for Association

## 2.6 Related Works

Having described our general biomedical relation extraction framework, we will discuss existing approaches to the relation extractions task. Significant research [56–58] has been reported for automatic relation extraction between biological entities (e.g. gene-disease, drug-drug). These approaches to relation extraction can be categorized into three kinds of methods: co-occurrence based, machine learning, and rule-based. Co-occurrence based methods such as disease-drug [59], mutation-disease [60] use the co-occurrence of entities in text to determine a relation.

Machine learning-based approaches use annotated data to learn models and view relation extraction as a classification problem. Different machine learning algorithms have been used for relation extraction including Support Vector Machines, Naive Bayes, logistic regression, etc. Machine learning methods for different relations such as event extraction [57,61–63], protein-protein interaction (PPI) [1,2,4,64–69], chemical-disease relation [35,70–72] etc. have been reported. Machine learning-based systems' effectiveness highly depends on the training set size and quality. For certain specific biomedical relation tasks, such as PPI, a sufficient amount of annotated corpora is available [68]. There is a lack of annotated corpora for general biomedical relation tasks. Constructing such training sets is time-consuming and requires domain expertise.

Rule/pattern-based methods instead of relying on annotated corpora require linguistics and domain knowledge to develop rules/patterns to extract relations from text. These rules are hand-crafted based on plain text and/or syntactic structure. Rule-based systems define patterns based on plain text or shallow parser output have been reported for biological event extraction [73], PPI extraction [74], protein phosphorylation information [52], and mutation-disease association [9]. Other rule-

based system based on deep parsing to develop rules for event extraction [75], regulatory event extraction [76], and microRNA-gene regulation relation [26].Sentence simplification approaches have also been studied to assist relation extraction [74,77,78].

Note that all the above works on relation extraction (machine learning and rule-based) focus on specific relations between biological entities and concepts. Our framework described in the above sections aims at identifying and extracting biomedical relations connecting different entities and concepts in a general way rather than focusing on a specific type of relation. Also, our relation extraction system is modular and extendible and provides an easy-to-use convention to write patterns to add more rules if needed. Our framework is built using general linguistic principles rather than domain knowledge. Different applications focusing on specific biomedical relations have and can be developed using our framework such as those above or those we developed described in the next three chapters.

The different rules/patterns developed for extracting our CAIR relations are motivated by linguistic theories, such as Lexicalized Tree Adjoining Grammar (LTAG) [36,79]. Additionally, ideas discussed in VerbNet [80,81], and FrameNet [55], verb subcategorization [82,83] were also exploited when developing our rules for relation extraction.

## 2.7 Conclusion

We developed a framework for relation extraction to facilitate the development of different biomedical relation extraction systems by handling certain common aspects of such RE systems, such as sentence splitting, parsing, syntactic dependency conversion, and handling common linguistic constructs (is-a, null arguments). The

major contribution of this system is its generality. Most of the existing relation extraction tools in the biomedical domain focus on specific types of relations such as protein-protein interaction, gene target relations, etc. This relation extraction framework can be used to quickly develop applications to serve different information needs. We have shown how this relation extraction framework can be used to extract CAIR (Connections through Association, Involvement, and Regulation) relations, which forms the basis of the different text-mining tools described in this dissertation.

# Chapter 3

# EXTRACTING THE ROLE OF MICRORNAS IN DISEASE

## 3.1 Introduction

MicroRNAs (miRs) are increasingly being recognized as critical players in human diseases, and questions concerning the role of microRNAs arise in many areas of biomedical research. The role of miRs in cancer is very well established, with a wealth of studies demonstrating the participation of microRNAs (miRs) in multiple cancer-related processes in diverse tissue types [84]. The amount of miRNA-related literature is increasing rapidly, which makes it difficult for researchers and curators to keep up to date. There are several manually curated databases of microRNA-disease associations gathered from the biomedical literature; however, it is difficult for curators of these databases to keep up with the explosion of publications in the microRNA-disease field. Figure 1.1 (in Chapter 1) showed this growing trend in the numbers of microRNA and disease related articles published between 2000 and 2019. In 2019 alone, there were 14,826 microRNA and disease-related publications, which was a 15% increase from 2018 and a 32% increase from 2017. Moreover, automated literature mining tools that assist manual curation of microRNA-disease associations currently capture only microRNA expression in the context of cancer. Thus, there is a clear need to develop more sophisticated automated literature mining tools that capture a variety of microRNA properties and relations in the context of multiple diseases to

provide researchers with fast access to the most recent published information and to streamline and accelerate manual curation.

**Types of miR-disease connections:** Although at a mechanistic level, miRs influence disease through their effects on the expression of their target genes, in the scientific literature miRs are associated with diseases through a variety of relationships. In some cases, miRs are directly connected with the disease itself or sometimes they are connected to a feature or outcome of the disease, such as aggressiveness [85], invasiveness [86], or patient survival [87]. miRs can also be linked to biological processes that are, in turn, connected to the disease. This category of processes includes ones such as apoptosis [88], or cholesterol transport [89]. In other cases, miRs are identified as biomarkers [90] or therapeutic targets [91].

A survey of the literature for the various types of miR-disease associations resulted in the following most common types of connections as exemplified in Examples 1-4 below. In these examples, the microRNA and the disease are bolded, while disease aspect (process, outcome, biomarker), which indicates role of the miR in the disease in underlined.

1. miR connected to Disease Outcome (poor prognosis, survival, etc) as in Example 1.
2. miR connected to Disease/Cellular Process (e.g., apoptosis) as in Example 2.
3. miR is a therapeutic target and biomarker for disease as in Example 3.
4. miR expression in disease tissues/cell/cell-line etc. as in Example 4.

Example 1: low expression of **miR-449a** was highly correlated with <u>cancer recurrence</u> and <u>survival</u> of **lung cancer** patients. [PMID: 24211326]

Example 2: Overexpression of **miR-204-3p** enhanced **glioma** cell apoptosis. [PMID: 27487563]

Example 3: Serum **microRNA-145** as a novel biomarker in **human ovarian cancer**. [PMID: 25722112]

Example 4: **MiR-224** is overexpressed in **human gastric cancer**. [PMID: 24796455]

miR's connections to disease processes or outcomes are typically through one of CAIR relations of Involvement, Regulation, and Association as in Examples 1 and 2. Connections to a disease via biomarker or therapeutic target is through is-a relations as in Example 3, while the expression level of miR in disease is stated through found-in and comparison relations as in Example 4. Note, the first three (1-3) miR-disease association also conveys additional information indicating the miR has a role in the disease. This might not be true for connection 4 (expression) as an expression in disease does not necessarily imply that the miR has a role in the disease but might be implicit.

In this chapter, we cover the first aspect of our work to extract **miR-disease associations**. This aspect involves extracting the **role of miRs** in diseases through connections 1-3 (disease outcome, process, target, and biomarker), which forms the basis of a text-mining tool miRiaD [23]. **miRiaD** (microRNAs in association with Disease) is a text-mining tool that automatically extracts associations between microRNAs and diseases from the literature, indicating the role of microRNAs in diseases. The second aspect of miR-disease association through connection 4, which involves extracting the **expression level information of microRNAs** in disease tissues/cells and not necessarily indicating a role in disease will be presented in chapter 6. Section 3.2 explores related works: curated databases and various text-

mining tools. In Section 3.3 we present the methodology behind the development of miRiaD and finally in Section 3.4, we present an evaluation of the system.

## 3.2 Related Works

***Curated Databases:*** There are currently several high-quality databases that capture miR-disease connections and some of the above relations, such as miR2Disease [20], miRCancer [21], and the Human microRNA Disease Database (HMDD; [22]). These resources are literature-based and support searches for miR or disease of interest. miR2Disease and miRCancer provide information on miR expression in disease; miR2Disease additionally covers miR target genes in the context of a disease. HMDD documents miRs that are potential biomarkers and provides several analysis tools, such as miR enrichment analysis. miR2Disease and HMDD are manually curated; thus they are limited by the time-consuming nature of manual curation and have difficulty keeping up with the explosion of publications in the miR-disease field. For miR expression in disease, in addition to miR2Disease, dbDEMC [92,93] is a database of differentially expressed miRNAs in human cancers obtained from de-novo analysis of high-throughput expression data such as microarrays.

***Text-mining tools:*** Automated literature mining tools could help streamline and accelerate the curation process as well as provide researchers with fast access to the most recent published information; however, currently, such tools are limited and have not been widely adopted. Most of the miR-related literature mining tools available focus on the extraction of miR-target gene relations without regard to disease, and rely on relatively simple text mining techniques, such as co-occurrence of miR and disease in the same sentence or abstract. These include miRSel [27] and the

tools used by the miR-target databases miRWalk [94], TarBase [95], and miRTarBase [96]. miRCancer [21], is one of the few resources that uses a rule-based system to identify disease-relevant miR information in the literature (further confirmed by curators), but is limited to detecting miR expression associations in cancer.

Thus, we have developed a comprehensive system for miR-disease connections resource, which automatically extracts from the biomedical literature associations between miRs and diseases together with any intermediate relations that bridge the association, thereby capturing various ways in which a miR plays a role in a disease.

## 3.3 Methods

As indicated before, miR's role in disease can be through connections of miRs with a disease concept (e.g. disease outcome, process) or through a biomarker and therapeutic target relation. In this section, we will describe the various aspects involved in developing miRiaD [23]. Firstly, we need to describe the different relations that will be used to connect a microRNA to a disease in Section 3.3.1 In Section 3.3.2 and 3.3.3, we present techniques to determine the type of connected arguments and infer the disease to build the miRiaD database.

### 3.3.1 Relations of Interest

#### 3.3.1.1 Connection to disease outcome/process: CAIR relations

As indicated before connections of microRNA to disease, disease process, and disease outcome are through **CAIR** relations. In general, miR or state of miR (e.g., differential expression, methylated state) can be connected to disease or disease-related concepts (e.g., disease outcome or process). For example, we can extract a

miR's involvement in the outcome of a disease (e.g. poor progression) or its role in the disease process (metastasis) for a disease.

Consider the sentences below in Example 5a-e. In Example 5a, the mir-26a state (downregulation) is related to the disease osteosarcoma, more specifically the disease process of tumor metastasis. Thus, we need to consider the **Association** relation between the miR and the disease process (tumor metastasis). It is evident from the sentence that this disease process is in the context of the disease osteosarcoma. Also, a **Regulation** relation can connect a miR to a disease process as in Example 5b. Sentence 5c is an example of a connection between a miR (microRNA-372) and a disease outcome (poor prognosis) through an Association Relation. Thus, when connecting a microRNA to disease, we consider all CAIR relations, which can connect a miR or miR expression of miR to a disease-related concept such as disease outcome/process. CAIR connections of miR to disease via null-arguments are presented in Examples 5d and 5e. As discussed in Chapter 3, these relations are Involvement, Regulation, and Association.

Example 5a: Downregulation of **mir-26a** is *associated with* tumor metastasis in **osteosarcoma**. [PMID: 24452597]

Example 5b: **MiR-30a-5p** *Suppresses* Tumor Metastasis of **Human Colorectal Cancer**. [PMID: 27576787]

Example 5c: **MicroRNA-372** is *associated with* poor prognosis in **colorectal cancer**. [PMID: 22456107]

Example 5d: **MicroRNA-9** promotes tumor metastasis via repressing E-cadherin in **esophageal squamous cell carcinoma**. [PMID: 25375090]

Example 5e: Tumor suppressive **miR-1** induces apoptosis through direct inhibition of SRSF9 in **bladder cancer**. [PMID: 22178073]

#### 3.3.1.2 Biomarker/Therapeutic Target: Is-a relation

miRs often serve as biomarkers or therapeutic targets for disease. Biomarker and therapy information is of particular importance to researchers studying miRs relationships to diseases. Example 6a and 6b convey such biomarker and therapy information. Such information is often found to be stated in literature through **Is-a** relations. Our EDG-based relation extraction framework will extract Is-a relations between miR-21 and diagnostic marker in Example 6a and therapeutic target in Example 6b. Note the diseases (gastric cancer and colorectal cancer) are both in the second argument of the relation attached through prepositions.

Example 6a: Circulating **MicroRNA-21** Is a Potential Diagnostic Biomarker in **Gastric Cancer**. [PMID: 26063956]

Example 6b: **MiRNA-21** may serve as a novel therapeutic target in **colorectal cancer** (**CRC**). [PMID: 25603978]

### 3.3.2 Determining Argument Types

We need to detect different entities and concepts to build a miR-disease role database. Firstly, we need to detect the mentions of miRs. Although miRs are mentioned in text in a variety of ways (e.g., miR-1, microRNA1, miRNA-1, let-1, etc.), they follow a well-established naming convention as described in miRBase [97]. We detect such miR mentions by using simple regular expressions. Currently, the

system does not differentiate between different forms such as miR-1a, miR-1-3p, and miR-1.

Additionally, the first argument of the CAIR or is-a relation connecting the miR to a disease concept can be a "state" (modifiers) of the miR. For the states of the miR, we only consider its expression or mutation status. We detect this by searching for noun phrases headed by trigger words such as "level", "expression", "mutation", "variants" or "polymorphism" etc. that modify the miR i.e. attached through a *compound* edge (e.g. miR-21 expression) or a *nmod:of* edge (e.g. "expression of miR-21").

We use a dictionary-based approach to detect locate **disease-related concepts** such as disease process, outcome, and diagnosis. We have compiled a dictionary by examining hundreds of sentences from the literature. However, for disease mentions, we use the Pubtator [2][98] database, which records the disease mentions tagged in Medline abstracts by NER tool DNorm [99].

### 3.3.3 Determining the Associated Disease

In all examples above, we associated the disease with the miRNA because the disease mention occurred in the same sentence that indicated the miR-disease connection. In most of these cases, the disease was in the noun phrase corresponding to the argument of the general relations or attached to it by a prepositional phrase. However, this is not always the case and the disease that must be associated might

---

[2] We use PubTator since it is a precompiled database containing bio-entity annotations for entire Medline and is updated regularly.

only be present elsewhere in the same abstract. In these cases, the sentence either mentions generic disease terms such as tumor, cancer, disease or disease-related concepts such as disease outcome (poor survival), disease process (metastasis), patients, etc. where the specific disease being referred to will be mentioned somewhere else in the abstract. For example, the generic disease term in Example 8 is "tumor metastasis". The disease being referred to here is gastric cancer, which is mentioned in several places in the abstract including the title. Thus, there is a need to infer the disease in such cases.

> Example 8: A high plasma level of miR-15b-5p was correlated with distant **tumor metastasis**. [PMID: 28560431]

We use the principles of Patient Context reported by Mahmood et al. [9] to infer the referred disease. Notice some abstracts contain experimental studies sentences mentioning the patient or patient sample used in the disease study. Mahmood et al. identifies such sentences (Patient Mention sentences) and notes that the disease central to the abstract is often mentioned in these sentences. The authors define Patient Context (PC) sentence if it is the first Patient Mention sentence in the abstract. We use the Patient Context sentence to determine the associated disease. If no Patient Context sentence can be detected, we use the title and if no disease is mentioned in the title, we consider the first sentence of the abstract to refer to the associated disease.

### 3.3.4 miRiaD Database

Having determined the type of relations relevant for miR-disease relationships, associated disease, and argument types we can now build a comprehensive miR-disease connection database indicating a role in disease. miRiaD was applied to the

entire Medline corpus, identifying 8301 PMIDs with miR-disease associations. The different components of the miRiaD database include (i) microRNA, (ii) disease (with normalization information) (iii) relation type (CAIR, is-a), (iv) connected argument and its type (e.g. disease outcome/process, biomarker etc.) (v) text evidence.

We have also developed a preliminary website for an interactive query of miRiaD miR-disease association extraction. The interface accepts PubMed-like queries as input, thus supporting queries like a miR name, or a disease name, or any biological concept. The interface is available at the URL[3] below. More details about the interface can be found in our miRiaD paper [23].

### 3.4 Evaluation

We evaluated miRiaD's effectiveness in extracting the role of miR in disease associations through our relations extraction framework. As noted earlier the disease might be in a different sentence than the sentence where the miR-disease relationship is stated. In this evaluation, we limited our evaluation to sentences where both the microRNA and disease co-occur to ease the burden on the annotators. For this study, we evaluated miRiaD with respect to the extraction of a range of relations that appear in text connecting a miR to a disease that may be of interest to biomedical researchers. The annotator, a bioinformatics scientist with a doctorate in cell biology, was asked to mark all the relations indicating a miR-disease association in a sentence, such that we can assess miRiaD's ability to detect such relations. The annotator was not involved in

[3] http://biotm.cis.udel.edu/miRiaD

the development of the system. In the subsection 3.4.1 below, we will the describe the dataset creation and annotation process.

### 3.4.1 Experimental Setup

We randomly selected a set of 200 sentences from Medline abstracts, where each sentence had a mention of a miR and disease. The annotator was asked to mark the sentences as relevant if they contained a relationship between a specified miR and disease (indicating the role of the miR in the disease) or not relevant (indicating no relation between the miR and the disease). Out of the 200 sentences, 189 were marked as relevant and 11 were marked as not relevant. In addition, the annotator marked all relations in the sentence indicating an association between a miR and a disease, including miR and disease-related concepts. A relation annotation involves marking the trigger/predicate and the arguments of the predicate that indicates the connection between the miR and the disease/disease process. For example, in a positive association between **mir-26a** and the disease "**osteosarcoma**" as exemplified in Example 5a, the annotator marked the following the relation: "associated with" (predicate of the relation), "Downregulation of mir-26a" (argument of the relation), "tumor metastasis in osteosarcoma" (argument of the relation).

Example 5a: Downregulation of **mir-26a** is *associated with* tumor metastasis in **osteosarcoma**. [PMID: 24452597]

This relation was compared with the output of CAIR relation extraction system by another annotator, a computer scientist with extensive experience in the field of BioNLP. Because multiple miRs, diseases, and types of relationships could be found

within the same sentence, the annotations yielded a total of 334 relations from 189 relevant sentences.

### 3.4.2 Results

The results of this evaluation are shown in Table 1. Out of the 189 relevant sentences annotated, miRiaD was able to extract relations from 181 of them hence marking them relevant. (181 TP and 8 FN). Out of the 11 annotated not-relevant sentences, miRiaD did not extract any relation for 8 relevant sentences (8 TN). Additionally, miRiaD identified relevant relations in 3 sentences annotated as not-relevant (3 FP). Row 1 of Table 1 shows the evaluation results based on sentences.

Further, we also present evaluation results based on the annotated relations. For calculation of True Positive (TP), False Negative (FN), and False Positive (FP) we computed matches between the annotated relations with the relations extracted by miRiaD. As indicated earlier, this matching was done by another annotator, a computer scientist with extensive experience in the field of BioNLP. A TP was assigned if all arguments and the trigger extracted by the RE system matched the annotated arguments and trigger as determined by the second annotator. A mismatch either the predicate trigger or any argument resulted in both a false positive and false negative, indicating miRiaD missed an annotated relation (FN) and outputted an incorrect relevant relation (FP). This yielded a recall of 84.1%, a precision of 95.5%, and an f-score of 89.4% (281 TP, 53 FN, and 13 FP) as shown in Row 2 of Table 1.

| | TP | FN | TN | FP | Recall | Precision | F-score |
|---|---|---|---|---|---|---|---|
| Sentence-Based | 181 | 8 | 8 | 3 | 95.7 | 98.3 | 96.9 |
| Relations-Based | 281 | 53 | 8 | 13 | 84.1 | 95.5 | 89.4 |

Table 3.1: miRiaD Evaluation Results

### 3.5 Conclusion

We have developed **miRiaD** (microRNAs in association with Disease), a text-mining tool that automatically extracts associations between microRNAs and diseases from the literature, indicating the role of microRNAs in diseases. miRiaD is a direct application of CAIR relations. There are several ways in which a microRNA's role in disease is stated. For example, a microRNA can influence disease through its effect on the expression of its target genes, can be associated with the outcome of a disease, can serve as a biomarker, or can play a role as a therapeutic target for disease.

miRiaD attempts to extract the "myriad" ways the microRNA and disease are connected. Specifically, miRiaD extracts various types of miR-disease associations such as (1) miR associations to disease outcome (poor prognosis, survival, etc.), (2) miR association to disease or cellular processes (apoptosis, metastasis, etc.), and (3) miR connections to disease via biomarker and therapeutic target relations. Additionally, miRiaD's results have been used by other people to create a miRNA resource called emiRIT [34]. emiRIT is an informatics portal with mined microRNAs

in biological networks that incorporates text mined results from miRiaD. miRiaD was applied on all MEDLINE abstracts till May 2020 and text-mined results were integrated into emiRIT. emiRIT contains information about 3,099 microRNAs, 255 diseases and 12,300 microRNA-disease associations from 121,371 abstracts.

# Chapter 4

# IMPACT OF PROTEIN PHOSPHORYLATION

## 4.1 Introduction

A central theme of this dissertation is the development of a relation extraction system to extract CAIR relations (Connections through Association, Involvement, and Regulation) to connect entities and biological concepts. In the previous chapter, we described a direct application (miRiaD) of CAIR relations, which extracted the role of miRs in diseases. To demonstrate the wide applicability of CAIR, we present another application of CAIR relations in this chapter: a text-mining tool to extract the impact of protein phosphorylation. In the subsequent subsections, we discuss the motivation behind extracting the phosphorylation impact and describe the task.

### 4.1.1 Motivation

Post-translational modifications (PTMs) are chemical modifications of amino acid residues (site) of proteins (substrate) by a catalyst protein (enzyme). PTMs play an important role in the regulation of function, activity, and location of a wide range of proteins. Phosphorylation is one of the most common forms of PTM. Phosphorylation of proteins by kinases are responsible for the activation and deactivation of many critical cellular pathways such as regulatory mechanisms of metabolism, cell division, cell growth, and differentiation.

Often protein phosphorylation has functional implications on the substrate such as leading to either alternative subcellular localization of the protein and/or affecting

the interaction with distinct binding partners. As an example, phosphorylation of the protein Smad2 can determine its interaction partners as illustrated in the sentence in Example 1 below. The phosphorylation and the interaction events are emphasized in bold, and the impact on the interaction in italics.

Example 1: TbetaRI **phosphorylation of Smad2** on Ser465 and Ser467 *is required for the* ***interaction between*** **Smad2** and **Smad4.**

The functional impact of phosphorylation on the substrate is not yet well represented in public databases [100]. However, this information is critical for the understanding of protein networks and the prediction of the functional outcomes. Thus, in this work, we present a text-mining tool to extract the functional impact of phosphorylation on the substrate such as its interaction with other proteins, its subcellular localization, and further post-translation modification of the substrate.

### 4.1.2 Task Definition

In this work, we extract the functional impact of phosphorylation on the three most common aspects of the substrate: (1) substrate's interaction with other proteins (binding partners), (2) alternative subcellular location of the substrate, (3) subsequent further post-translational modification (acetylation, ubiquitination, etc.) of the substrate, which will refer as **impacted events** henceforth.

We define the functional impact of the phosphorylation on the impacted event (1) if the impacted event is started or is "due to" phosphorylation, or (2) if the impacted event is positively or negatively regulated ("promoted", "blocked", "impaired" etc.) because of the phosphorylation event. Note, in both these cases there is a temporal ordering between the two events, where the impacted event follows the phosphorylation event. An example of the first case, where the impacted event "due

to" the phosphorylation is depicted in Example 1 above, where the phosphorylation of the "Smad2" causes the interaction of the phosphorylated protein (substrate) "Smad2" with another protein "Smad4". An example sentence of the second case is provided below in Example 2, where the phosphorylation of "PTEN" impairs its interaction with the binding partner "Cdh1".

> Example 2: Phosphorylation of **PTEN** on Ser-380 <u>impaired</u> **<u>its</u>** interaction with **Cdh1**. (PMID: 24811168)

In Section 4.2.1, we will describe the three impacted events with examples and indicate what we intend to extract for each one of them. In Section 4.2.2, we will discuss the different ways the connection between the phosphorylation event and the impacted event are stated in text that we consider for extracting of the functional impact of phosphorylation. In Section 4.2.3 we discuss the related works. The system itself and each component are described in detail in Section 4.3. Evaluation and analysis of results are presented in section 4.4.

## 4.2 Background

### 4.2.1 Type of impacted events

Below, we provide examples for each of the **three impacted events** of the substrate and indicate what we intend to extract for each one of them.

**(1) Protein-Protein Interaction (PPI):** In these cases, the phosphorylation of the substrate either causes or impacts the substrate's interaction with another protein (binding partner) as exemplified in Example 2. In Example 2, the phosphorylation of the protein "PTEN" at the site "Ser-380" impacts ("impairs") its interaction with the binding partner "Cdh1".

Example 2: Phosphorylation of PTEN on Ser-380 impaired its interaction with Cdh1. (PMID: 24811168)

In these cases, we will extract the substrate, the impact trigger, and the interacting proteins, where one of the interactants is the same as the substrate. Thus, from Example 2, we will extract the 3-tuple:

<PTEN (*substrate and PPI interactant 1* ), impaired (*impact trigger*), Cdh1 (*interactant 2*)>

Note, one of the interacting proteins (interactant 1) as stated in sentence (2) is "its", which will be resolved by our tool to "PTEN". Also, this interactant is the same as the phosphorylated protein. Note, the site of phosphorylation ("Ser-380" in this case) is not extracted by our tool, as a phosphorylation relation extraction tool such as Rule-based LIterature Mining System for Protein Phosphorylation (RLIMS-P) [52,53] exists that can extract the phosphorylation sites. Note, RLIMS-P only extracts phosphorylation information (substrate, kinase and site) and not the impact of phosphorylation, which is the focus of this chapter.

**(2) Subcellular Localization (SL):** Subcellular localization (SL) is a process whereby a protein is transported to, and/or maintained in, a specific location (subcellular location) within a cell including the localization to the cell membrane**.** Phosphorylation of proteins can also have an impact on this process of subcellular localization of the phosphorylated protein (substrate) as exemplified in Example 3. In this example, the phosphorylation of "Raf-1" causes its localization to the subcellular location "mitochondria".

Example 3: p21-activated Kinase 1 (Pak1)-dependent phosphorylation of Raf-1 regulates its mitochondrial localization (PMID: 15849194)

From this sentence in Example 3, we will extract the substrate (or the localized protein), the impact trigger, and the destination subcellular location as indicated in the 3-tuple below:

<Raf-1 (*substrate and localized protein*), regulates (*impact trigger*), mitochondria (*subcellular location*)>

Note, in these cases, the substrate moves from one subcellular location to a destination subcellular location. In this work, we only extract the final destination location as the initial location of the substrate is only stated in a small subset of these cases in text.

**(3) Post-translation Modification (PTM):** As indicated earlier, Post-translational modifications (PTMs) are chemical modifications of amino acid residues (site) of proteins (substrate) by a catalyst protein (enzyme). Phosphorylation can have an impact on subsequent PTM of the substrate as exemplified in Example 4. In this example, the phosphorylation of "PLK1" impacts ("inhibited") its further modification ("ubiquitination"). Further PTMs considered in this work are acetylation, glycosylation, and three ubiquitination-like modification, namely, ubiquitination, sumoylation, and neddylation.

Example 4: Phosphorylation of PLK1 by c-ABL inhibited PLK1 ubiquitination and degradation. (PMID: 27899378)

From this sentence in Example 3, we will extract the substrate (same as the substrate of the further modification), the impact trigger, and the further PTM event as indicated in the 3-tuple below:

<PLK1 (*substrate and substrate of further PTM*),), inhibited (*impact trigger*), ubiquitination (*further PTM event*)>

### 4.2.2 Type of Impact Connections

There are a variety of ways as stated in literature that connects a phosphorylation event to an impacted event (PPI, SL, or further PTM), which are described below.

**Type A (CAIR):** In this category, there is direct evidence of the phosphorylation event impacting and changing the substrate's property event as exemplified in our previous Examples 2-4. In Example 2, the phosphorylation of "PTEN" impacts (impairs) the interaction event of the substrate with its binding partner "Cdh1". We have found these connections are stated through CAIR relations (regulation, involvement, and association) in text.

> Example 2: Phosphorylation of PTEN on Ser-380 <u>impaired</u> its <u>interaction</u> with Cdh1. (PMID: 24811168)

**Type B (Temporal):** Sometimes the impact of phosphorylation is not made explicit by the author but its impact on the substrate appears to be implied. The authors connect the two events by specifying a temporal order between them: (1) phosphorylation event, and (2) the impacted event (PPI, PTM, localization), where the impacted event follows the phosphorylation event. There are two common ways this temporal ordering might be stated in the text.

In the **first** category, the temporal order is explicitly stated through time-related prepositions or verbs such as "upon, "following", "before", "via", "through" etc. as exemplified in Example 5a-c. For example in sentence 5a, phosphorylation of "Smad1" is followed by ("upon") the interaction between the substrate "Smad1" and its binding partner "Smad4".

> Example 5a: **Upon** phosphorylation by the BMP receptors, <u>Smad1</u> interacts with <u>Smad4</u>. (PMID: 10224145)

Example 5b: The inducible phosphorylation of IkappaBalpha is **followed** by its ubiquitination and degradation. (PMID: 9792644)

Note, in these cases, the presence of these explicit temporal triggers indicates an ordering of two events, where the impacted substrate's property event (PPI, SL, further PTM) follows the phosphorylation of substrate event. This notion of ordering indicates that the impacted event might be "due to" the phosphorylation event and thereby implying impact.

An alternate **second** way of impact due to temporality includes sentences, where the substrate (phosphorylated protein) is an **argument** of the substrate's impacted event (PPI, PTM, localization). Consider the sentence in Example (6), which indicates that in the phosphorylated state the protein "BLNK" is further modified (ubiquitinated). However, whether the phosphorylation has any impact on the ubiquitination itself (i.e. if BLNK is ubiquitinated regardless of its form, phosphorylated or non-phosphorylated) is less clear. But, there is a possible hint of impact since the impacted event occurs only after the phosphorylation event.

Example 5: Threonine 152-phosphorylated BLNK is ubiquitinated at lysine residues 37, 38, and 42. (PMID: 22334673)

### 4.2.3 Related works

Resources and tools that capture information about the impact of phosphorylation are limited. Curation efforts are underway in Protein Ontology (PRO) [100], which provides an ontological structure to capture information about protein classes, phosphorylated protein forms (proteoforms), and their properties such as protein-protein interactions (PPIs). eFIP (Extracting Functional Impact of Phosphorylation) [101,102] is a text-mining (TM) tool that identifies phosphorylated

proteins and phosphorylation-dependent PPIs. To the best of our knowledge eFIP is the only automated TM tool that extracts some form of impact of phosphorylation. eFIP is limited to impact on PPIs and does not extract phosphorylation impact on other substrate properties such as the impact of subcellular localization and further post-translational modification. While works for extracting the impact of phosphorylation as a whole is limited, other systems addressing individual tasks for extracting phosphorylation, PPI, subcellular localizations relations exist.

For the detection of phosphorylation relations in text, systems such as Rule-based LIterature Mining System for Protein Phosphorylation (RLIMS-P) [5,52,53], and MinePhos [7] has been developed. Phosphorylation events were also investigated by the BioNLP 2011 Shared task [8]. In our work, we use RLIMS-P to extract phosphorylation information, since it covers a wide variety of phosphorylation event statements, and has been recently improved [103] with increased performance. Several works towards extracting PPI [1–4,104], subcellular localization [13–16], and other post-translational modifications (acetylation, glycosylation, ubiquitination, sumoylation, and neddylation) [105–107] relations from text have been reported. However, for several reasons, we have developed our in-house RE systems to extract the impacted events: PPI, subcellular localization (SL), and other post-translational modification (PTM) relations based on our RE framework. The primary reason is that since one of the arguments of our impacted event is the phosphorylated protein and might not be mentioned in the impacted event clause, existing tools indicated above fail to extract these impacted relations as they expect two or more arguments. Another reason is to allow for easy integration with our pipeline. Different works for detecting and extracting relations between biological and medical events [108–111] have been

reported. Since we are interested in impact relations between phosphorylation events and various substrate property events (PPI, SL, PTM), these general approaches would have required extensive adaption to be applied to our case.

## 4.3 Methods

The extraction of the functional impact of phosphorylation on the substrate can be broken down into several different tasks. Since we are interested in the impact of phosphorylation, we will only process sentences that contain lexical triggers indicating a phosphorylation event. Thus, the initial step given a sentence is to determine whether there is a phosphorylation event in the sentence. This is done by checking for lexical triggers such as “phosphorylation”, “phosphorylated”, “phosphorylate” and “phosphorylate”.

After determining the presence of a phosphorylation event, the **first** step involves extracting the arguments of the impact event (if it exists), which will be different for each impacted event. For example, for impact on PPI, the arguments of the impacted event (PPI) will be the two interacting proteins. The **second** step is to determine and extract the connection between the phosphorylation event and the impacted event, which either indicates explicit impact (through CAIR relations) or implicit impact (through temporal ordering) as discussed in Section 4.2.2. The final **third** step involves the extraction of the phosphorylated protein i.e., the substrate. Please note that the phosphorylated protein (substrate) will be one of the arguments of the impacted event i.e. (1) one of the interactants of a PPI relation, (2) the localized protein, or (3) the further post-translational modified protein. The three steps: (1) Extracting arguments of the impacted event, (2) Connecting the phosphorylation event

to the impacted event, (3) Extraction of the phosphorylated protein (substrate) are described in sections 4.3.1 to 4.3.3, below.

### 4.3.1 Extracting arguments of the impacted event

As indicated earlier, in this work we are interested in extracting the impact of phosphorylation on three events: Protein-Protein interaction (PPI), Subcellular Localization (SL), and further Post-Translational Modification (PTM). The extraction of each of these impacted events can be treated as a predicate-argument relation extraction task, where the event will serve as the predicate indicated through different lexical triggers and with different arguments based on the impacted event. We will use our EDG-based relation extraction framework to extract such relations, which leverages lexico-syntactic patterns defined on the syntactic dependencies. As already mentioned in chapter 2, EDG not only considers syntactic dependencies between words in a sentence but also utilizes information beyond syntax to capture different dependencies, and hence the set of rules to extract impacted relations in the context of phosphorylation impact is small.

Below we provide examples of the lexico-syntactic patterns for extraction of the impacted event and its arguments. A full list of triggers can be found in Appendix C and lexico-syntactic patterns can be found in Appendix D.

#### 4.3.1.1 Extraction of Protein-Protein Interaction (PPI)

PPI involves predicate-argument relations, where typically two proteins are arguments of a binary relation. The predicate of PPI relations are verb-based triggers that require prepositional complements. Such PP complement verb phrases include “binds to/with”, “interacts with”, “associated with ”, “dimerizes with”, etc. (A full list

of triggers can be found in Appendix C.4). We use the dependencies provided by our EDG framework from these lexical triggers to extract the protein arguments.

Consider the example in Figure 4.1a, where we use the edges *nusbj* and *nmod:to* from the trigger "binds" to add *arg0* and *arg1*, respectively. These numbered argument edges arg0/arg1 points to the head token of the two interacting proteins. Note, since the PPI relation is symmetric the numbering of the argument edges is not important for the final extraction of the arguments but is provided to simplify our explanation of the different lexico-syntactic patterns below. Verb trigger with different subcategorization (prepositions) will have a different syntactic edge for *arg1*. For example for the lexical trigger "interacts with" we will use the *nmod:with* edge.

A nominalized verb form can also be used in these cases as in Figures 4.1b and 4.1c. In these cases, the argument corresponding to the *arg0* modifies the nominal lexical triggers (Interaction of X). In these cases, we will use edge "nmod:of" (as in Figure 4.1c) or the *compound* edge (e.g. "PTEN interaction with …) to add *arg0.* Note, since one of the interacting proteins is the substrate in the impact sentences we consider, it might be referred to "its" and not repeated in the interaction clause as in Figure 1c. In these cases, we use the *nmod:poss* to add the *arg0* edge.

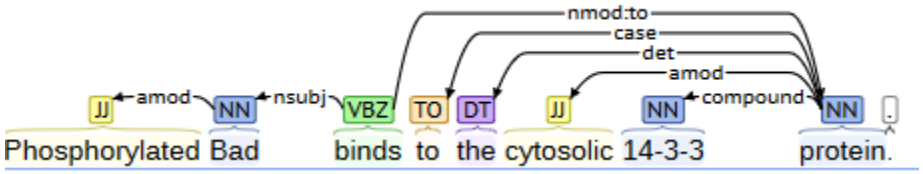


Figure 4.1a: PPI active verb form

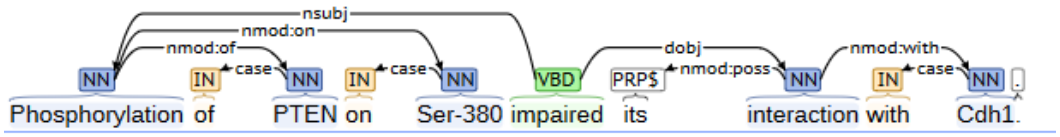


Figure 4.2b: PPI nominalized form 1

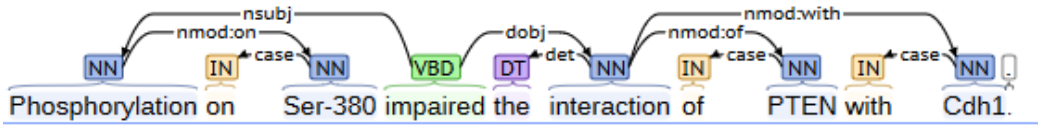


Figure 4.1c: PPI nominalized form 2

Traditional PPI relation extraction (RE) tools [1,3,4,64,65] extract PPI relations, where both interactants are mentioned in the interaction clause. However, as we have indicated earlier, one of the interactants in the impact sentences we consider is the phosphorylated protein i.e., the substrate and might not be mentioned in the interaction clause and the other interactant may be implicitly referred to from context. As our goal to extract the impact of phosphorylation and one of the interactants is the phosphorylated protein (substrate), we can infer the missing interactant in these cases as the substrate. This is the primary reason we have developed our own PPI RE system rather than using existing systems.

Consider the example in Figure 4.1d. Here the PPI relation is indicated by the trigger "interaction". The *nmod:with* "binding" gives only one interactant "Cdh1", but there is no other edge from "binding" to another interactant. Thus, in PPI relations, we add *arg0* and *arg1* independently of each other thus allowing for the addition of one numbered-argument edge if necessary. Hence, in this case, we will only add an *arg1*

by following the *nmod:with* edge from “interaction”. As the phosphorylation of “PTEN” impacts (“impairs”) the interaction, the other interactant is “PTEN”, which is the phosphorylated protein i.e. the substrate. The extraction of the substrate from the phosphorylation event is described in Section 4.3.3.

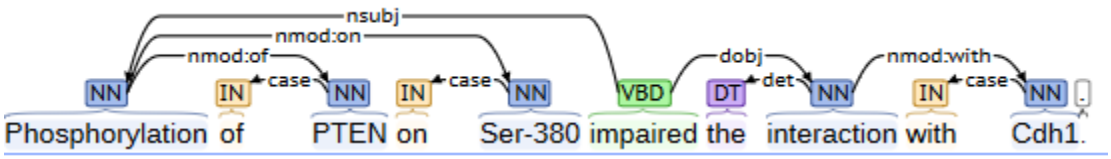


Figure 4.1d: PPI example with one interactant in interaction clause

Above, we have provided examples of some lexico-syntactic patterns to extract arguments of PPI relations. A full list of other syntactic patterns (e.g. X forms a complex with Y, X recruits Y, etc.) can be found in Appendix D.

#### 4.3.1.2 Extraction of Subcellular Localization

Subcellular localization (SL) involves predicate-argument relations, where a protein and final subcellular location (“nucleus”, “cytoplasm”, “mitochondria”) are arguments of a binary relation. The predicate of an SL relation are verb-based triggers such as “localized”, “translocated”, “trafficked”, “transported”, “imported”, “and exported” etc. with prepositional complements “to”, “at”, “in” and “into”. (A full list of triggers can be found in Appendix C.5). We use the dependencies provided by our EDG framework from these lexical triggers to extract the protein and subcellular location arguments.

Consider the example in Figure 4.2a, where we use the edge *nusbj* from the trigger "translocates" to add the *arg0* to the protein argument (STAT3) and the *nmod:to* edge to add the arg1 edge to the subcellular location (cytoplasm). Verb trigger with different subcategorization (prepositions) will have different syntactic edge for *arg1* subcellular location argument. For example, for the lexical trigger "accumulated at" we will use the *nmod:at* edge to add the *arg1* edge.

As with PPI, a nominalized verb form can also be used in these cases as in Figures 4.2b and 4.2c. Similar to PPI, here the *arg0* edge referring to the localized protein/substrate is added by following *nmod:of*, *compound*, or *nmod:poss* from the trigger. For example, we use the *nmod:poss* edge from "import" (as in Figure 4.2b) and *compound* edge from "localization" to add the *arg0* edge.

Notice in Figure 4.2b and 4.2c, the subcellular location (nucleus) is in the adjectival form ("nuclear") and *arg1* edge referring to the subcellular location is added following the modifier *amod* edge.

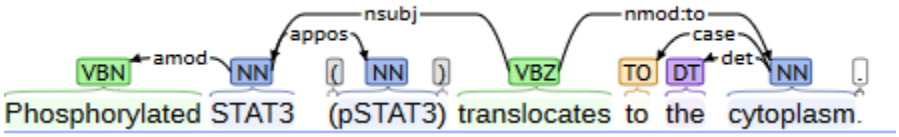


Figure 4.2a: Subcellular localization active verb form

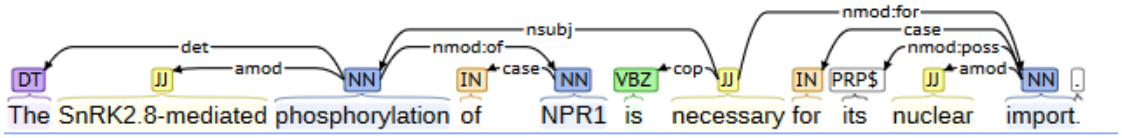


Figure 4.2b: Subcellular localization nominalized form 1

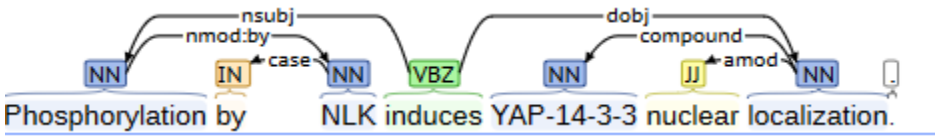


Figure 4.2c: Subcellular localization nominalized form 1

#### 4.3.1.3 Extraction of Post-translational modification (PTM)

As indicated earlier the further PTMs considered in this work are acetylation, glycosylation, and three ubiquitination-like modification, namely, ubiquitination, sumoylation, and neddylation. PTMs are chemical modifications of amino acid residues (site) of proteins (substrate) by a catalyst protein (enzyme) and thus have three arguments. But in this work, we are only interested in the further post-translated protein i.e. the substrate, which is the same as the substrate of the first phosphorylation event.

Thus, in our case, PTM relations involve predicate-argument relations, where the substrate protein will be the argument of a unary relation indicated by a lexical trigger. These lexical triggers indicating a PTM relation are verb-based such as include “ubiquitinate”, “acetylate”, “glycosylate”, “neddylate”, “sumoylate” etc. or their textual variations. We use the dependencies provided by our EDG framework from these lexical triggers to extract the substrate argument.

Here we use verb-based rules, specifically the passive and nominalized forms to the lexical triggers. Consider the example in Figure 4.3a, where we use the edge *nsubjpass* from the trigger “ubiquitinated” to add the *arg0* referring to the further modified protein. As with PPI and SL relations, in the nominalized form of a trigger we use the *nmod:of*, *compound*, or *nmod:poss* from the trigger to add the arg0 edge to the protein argument, which is the substrate of both the impacting phosphorylation

event and the further post-translational modification. For example, we use the *compound* edge (as in Figure 4.3b) and *nmod:poss* edge from "ubiquitination" (as in Figure 4.3c) to add the *arg0* edges.

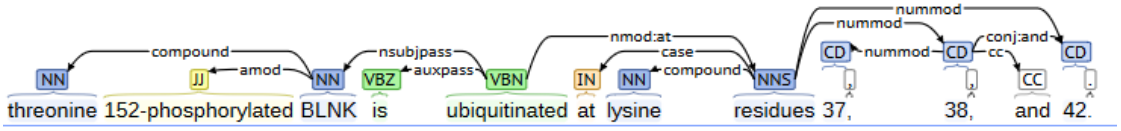


Figure 4.3a: Further PTM active form

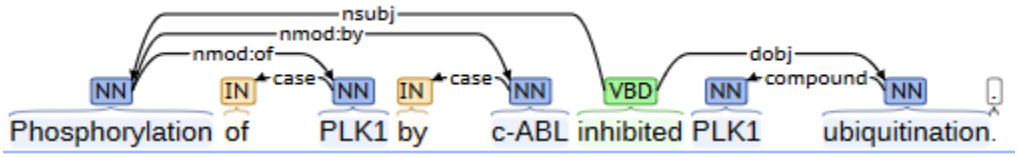


Figure 4.3b: Further PTM nominalized form

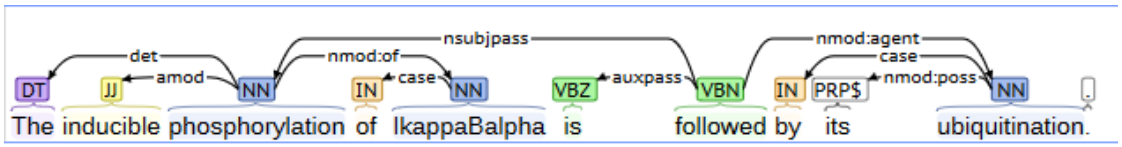


Figure 4.3c: Further PTM nominalized form

### 4.3.2 Connecting the phosphorylation event and impact event

As indicated in section 4.1.3, there are a variety of statements in the literature indicating the impact of protein phosphorylation on the substrate. We treat the extraction of impact on the substrate as relation extraction, where a protein phosphorylation event is connected to the impacted protein event via a lexical impact

trigger. As with the extraction of impacted event relations, we will use our EDG-based relation extraction framework to extract such impact relations, which leverages lexico-syntactic patterns defined on the syntactic dependencies. Below we provide examples of the most lexico-syntactic patterns for extraction of the impact relations. A full list of lexico-syntactic patterns can be found in Appendix D.

#### 4.3.2.1 Connections through CAIR relations: Type A

Type A impact relations are essentially CAIR relations discussed in Chapter 2. These impact relations involve predicate-argument relations, where a phosphorylation event and an impacted event (PPI, subcellular localization, and PTM) are the first and second arguments of a CAIR (regulation, involvement, or association) relation respectively. We will use our EDG-based CAIR system described in Chapter 2 to extract such impact relations. In these relations, a phosphorylation trigger ("phosphorylation") is the head of the phosphorylation event argument, and an impacted event trigger ("interaction", "translocation", "ubiquitination") as the head of the impacted event. Examples of such impact relations after applying our CAIR RE system are shown in sentences 7a, 7b, and7 c below, where the head of the arguments are bolded, argument phrases enclosed in brackets, and the CAIR (regulation) trigger underlined.

Example 7a: [**Phosphorylation** on Ser-380]_arg0 <u>impaired</u> [the **interaction** of PTEN with Cdh1]_arg1.

Example 7b: [**Phosphorylation** of Raf-1]_arg0 <u>regulates</u> [its mitochondrial **localization]**_arg1.

Example 7c: [**Phosphorylation** of PLK1 by c-ABL]_arg0 <u>inhibited</u> [PLK1 **ubiquitination]**_arg1.

#### 4.3.2.2 Connections through temporal ordering: Type B

As indicated earlier there are two ways temporal ordering is stated in text. The **first** category involves sentences, where the **explicit temporal** order is stated through certain words such as “upon, “following”, “before”, “via”, “through”. In these cases, a temporal ordering is explicitly stated indicating an impact.

The **first** set of these explicit ordering connections involves cases, where a phosphorylation event is connected to an impacted protein event by temporal triggers such as “upon”, “following”, “before”, “after” etc., and its textual variations. In these relations, we consider the trigger as our predicate and the phosphorylation and impacted substrate events as its arguments.

Consider the sentence structure in Figure 4.4. We use the “nmod:upon” edge to connect the phosphorylation and interaction (headed by “interacts”) clauses. Next, we use the “case” edge from the head of the dependent clause (“phosphorylation”) to add argument edges from the preposition “upon”. In this case, we add an arg0_temporal edge to “phosphorylation” and arg1_temporal edge to “interacts” thereby capturing the ordering of the event. Note this order is based on the lexical type of prepositional trigger in question as the ordering will be reversed if the temporal preposition was “before”. A full list of lexico-syntactic patterns for connections explicit temporal triggers can be found in Appendix D.

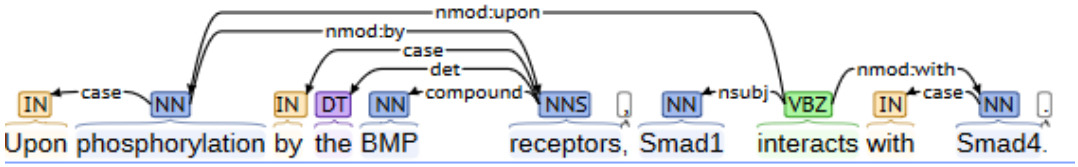


Figure 4.4: Explicit temporal impact example

The **second** set of temporal ordering between two clauses are mentioned through null-argument structures that were discussed in Chapter 2. Consider the sentence structure in Figure 4.5. In this case, two clauses below are connected by the preposition "by".

Clause 1: NRG1 receptor ErbB2 directly **binds** Dock7

Clause 2: **phosphorylating** Tyr-1118

In Chapter 2, we have discussed how we handle missing argument cases in these null argument cases. Due to the syntax of null-argument cases, there is temporal ordering between the two clauses. In addition to indicating an ordering, these null argument cases also indicate an implicit impact relationship between the two clauses i.e. the Phosphorylating event in clause 1 "causes" the PPI relation ("binding") in clause 2. Thus, we add argument edges to the head of the two clauses with the preposition "by" as the predicate trigger. We use the "advcl" edge to connect the phosphorylation and interaction (headed by "binds") clauses. Next, we use the "mark" edge from the head of the dependent clause ("phosphorylating") to add argument edges from the preposition "by". In this case, we add an arg0_temporal edge to "phosphorylating" and arg1_temporal edge to "binds" thereby capturing the ordering of the events.

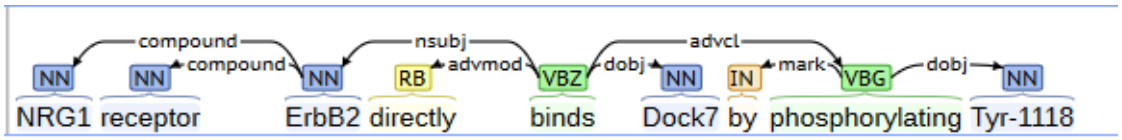


Figure 4.5: Impact through null-argument structures

The **third** case of temporal ordering involves sentences, where the phosphorylated protein (substrate) is an **argument** of the impacted event (further modification, interaction, or localization). Thus, these relations involve binary relations where the substrate is one of the arguments of an impacted event trigger. The second argument is either the second interactant, subcellular location, or a further PTM based on the corresponding impacted event as indicated in Examples 8a (PPI relation) and 8b (subcellular localization relation), where the head of the arguments are underlined, argument phrases enclosed in brackets and the impacted lexical trigger bolded. Since the impact in these cases is implied the impact trigger is absent.

Example 8a: [Phosphorylated Bad]_arg0 **binds to** [the cytosolic 14-3-3 protein]_arg1 (PMID 11526496)

Example 8b: [Phosphorylated STAT3 (pSTAT3)]_arg0 dimerizes and **translocates** to the nucleus. (PMID: 21294636)

The extraction of these binary relations is the same as "Extraction arguments of the impacted event" discussed in Section 4.4.1. The only additional constraint placed is that we only consider impacted event lexical triggers in the verb form (Examples 8a, 8b) and not in the nominalized form as in Example 9 below. Note that in Example 8a and 8b, where the trigger is in the verb form, the phosphorylation impact on the substrate appears to be implied. This implied "impact" of phosphorylation on the substrate is less clear in argument cases, where the lexical trigger is the nominalized form ("interaction"). This was determined in consultation with a Biology expert when developing the impact module.

Example 9: **Interaction** between serine phosphorylated IRS-1 and beta1-integrin affects the stability of neuronal processes.

### 4.3.3 Extraction of phosphorylated protein

Once we have extracted the impact relation between a phosphorylation event clause and an impacted event clause, we need to extract the phosphorylated protein i.e. the substrate. We use RLIMS-P [52,53] to extract the substrate. RLIMS-P is a system designed to extract protein phosphorylation information from text. It extracts the three objects involved in this process: the protein kinase, the phosphorylated protein (substrate), and the phosphorylation site (residue or position being phosphorylated). An example of information extracted by RLIMS-P is shown highlighted in Example 10. In these cases, we use the substrate extracted detected in the phosphorylation event argument ("**TbetaRI phosphorylation** of **Smad2** on **Ser465** and **Ser467"**) of the impact relation by RLIMS-P.

> Example 10: [**TbetaRI**]$_{\_kinase}$ [**phosphorylation**] of [**Smad2**]$_{\_phosphorylated\_protein}$ on [**Ser465** and **Ser467**]$_{\_site}$ is required for Smad2-Smad4 complex formation and signaling. (PMID 9346908)

Note, in some cases, the substrate might not be extracted by RLIMS-P. These include cases where the substrate is absent in the phosphorylation event clause as in Example 11. Here the first argument of impact ("Phosphorylation by NLK") does not contain the substrate. In these cases, the substrate is typically the protein argument of the protein event, which is "YAP-14-3-3" in this case. Thus, we infer the first protein interactant in PPI relations, the localized protein the subcellular location, and the further modified protein in PTM relations as the substrate if the RLIMS-P cannot detect it from the phosphorylation event argument.

> Example 11: <u>Phosphorylation by NLK</u> induces **YAP-14-3-3** nuclear localization.

### 4.3.4 Anaphora resolution

In some cases, the protein argument of the impacted protein event clause might just point to "its" as in Figures 4.1b, 4.2a, 4.3b. This pronoun needs to be resolved to its referring protein. Since in our case, the protein argument will always be the same as the substrate of the phosphorylation event, we can infer that the pronoun "its" in these cases refers to the substrate.

For example in Figure 4.1b, the argument protein edge points to "its", which refers to substrate "PTEN" in the phosphorylation clause ("Phosphorylation of PTEN.."), which we will infer when connecting the impact of the phosphorylation event and PPI event.

### 4.3.5 Entity Detection

In our different relation extraction tasks describe in section 4.4.1, we have indicated that arguments of the relations should be of certain types: proteins and subcellular location. We use PubTator [98] to typing proteins, which are normalized to NCBI Gene IDs. For typing subcellular location arguments we downloaded all 545 subcellular location terms from the UniProt website [112] and created a dictionary. Further since in some cases, the subcellular location can be mentioned in the text in the adjectival form (e.g. nuclear, cytoplasmic, and mitochondrial"), which are not included in UniProt, we extracted all the head words ("nucleus", "cytoplasm", "mitochondria") in the subcellular dictionary and added adjectival forms of the head words manually.

## 4.4 Evaluation

### 4.4.1 Experimental Setup

To evaluate our functional impact of phosphorylation on the substrate, we needed to have annotated datasets that annotated the elements extracted by our system.

For impact on **PPI**, we needed to have a dataset that annotates (1) the phosphorylated protein (substrate) or the PPI interactant 1 (2) impact trigger (if any), and (3) PPI interactant 2. To create a dataset for annotation, we downloaded the IntAct database [113], which contains manually curated 21,639 abstracts with 1063382 interaction relations. Each interaction relation in IntAct also contains a column, which indicates whether either interactant is phosphorylated. Filtering based on this column resulted in 2053 interaction relations. From this, we randomly selected 100 abstracts with 131 interaction relations. For each interaction relation, we asked a Bio-curator to first indicate whether the PPI relations was due to the impact of phosphorylation or not. If the interaction relation was relevant, we further asked the curator to annotate the 3-tuple (1) the phosphorylated protein / PPI interactant 1, (2) impact trigger (if any), and (3) PPI interactant 2. Additionally, if a PPI relation due to phosphorylation impact was missed by IntAct we asked the curator to annotate them too. We will refer to this dataset as P2PPI.

To evaluate our system on the impact on **subcellular localization**, we needed to have an annotated dataset that annotates a 3-tuple that contains: 1) the phosphorylated protein (substrate) / localized protein (same as the substrate), (2) impact trigger (if any), (3) Subcellular location. As this is a new direction of research, we could not find any existing resource that provides us with this type of annotation. To develop this dataset, we started with the RLIMS-P database, which contains

phosphorylation information extracted from all MEDLINE abstracts. From this database, we randomly selected 100 abstracts, which at least had one sentence mentioning a phosphorylation trigger and a subcellular location (refer to 4.3.5 for detection subcellular location). We then asked a Bio-Curator to annotate the abstracts with the 3-tuples indicating an impact of phosphorylation on subcellular localization. Abstracts with no tuples annotated were treated as not relevant i.e. negative cases. We will refer to this dataset as P2LOC.

To evaluate our system on the impact on **further post-translational modification,** we needed to have an annotated dataset that annotates a 3-tuple that contains: 1) the phosphorylated protein (substrate) / further PTM protein (same as the substrate), (2) impact trigger (if any), (3) type of the further PTM (ubiquitination, acetylation, etc.). As with subcellular localization, we could not find any existing resource that provides us with this type of annotation. Thus, to develop this dataset, we again started with the RLIMS-P database, which contains phosphorylation information extracted from all MEDLINE abstracts. From this database, we randomly selected 100 abstracts, which at least had one sentence mentioning a phosphorylation trigger and another PTM trigger. We then asked a Bio-Curator to annotate the abstracts with the 3-tuples indicating an impact of phosphorylation on PTM. Abstracts with no tuples annotated were treated as not relevant i.e. negative cases. We will refer to this dataset as P2PTM.

We evaluated our system on each of the P2PPI, P2LOC, and P2PTM datasets to see how well our system performs on each of our phosphorylation impact tasks. A true positive was assigned if all the 3-tuples for each respective dataset were extracted correctly by our system. A mismatch in any component of the 3-tuple resulted in both

a false positive (indicating an incorrect extraction) and a false negative (indicating our tool failed to extract the annotated 3-tuple).

### 4.4.2 Results and Discussion

We counted true positives (TP), false positives (FP), and false negatives (FN), and used the standard information retrieval metrics of Precision (P), Recall (R), and F-score for performance evaluation, where P = TP/(TP+FP), R = TP/(TP+FN) and F = 2PR/(P+R).

The evaluation results of our systems on the different datasets are shown in Table 4.1. We achieved an F-score of 0.80, 0.82, and 0.81 for the P2PPI, P2LOC, and P2PTM datasets. Analysis of the false positives and false negatives indicates that some of the errors were caused due to erroneous parsing of the sentences, which resulted in either incorrect or missing arguments of the impact relations. Also, some impact relations and impacted protein events (PPI, Subcellular localization, and PTM) were not captured due to lack of some lexico-syntactic rules as in Example 13 below, where we missed the impact relation on the PPI. Some false-negative cases involved inferring the impact from multiple sentences, which is currently not handled by our system. For example, in sentence 14, the previous sentence in the abstract indicated that the protein "paxillin" was phosphorylated and combined with "subsequently" in the sentence in question, it is clear that subcellular localization ("redistributed") was due to phosphorylation.

Example 13: FYN-T selectively phosphorylates FYB providing a template for the recruitment of FYN-T and SLP-76 SH2 domain binding… [PMID: 10409671]

Example 14: Subsequently, paxillin was redistributed to the basolateral cytosol and was degraded. [PMID: 9918850]

| Task/Dataset | Precision | Recall | F-score |
|---|---|---|---|
| **P2PPI** | 0.93 | 0.70 | 0.80 |
| **P2LOC** | 0.95 | 0.72 | 0.82 |
| **P2PTM** | 0.90 | 0.74 | 0.81 |

Table 4.1: Evaluation results for the impact of phosphorylation

## 4.5 Conclusion

In this chapter, we have described a system for extracting the functional impact of phosphorylation on the (1) substrate's interaction with other proteins (binding partners), (2) alternative subcellular location of the substrate, (3) subsequent further post-translational modification (acetylation, ubiquitination, etc.) of the substrate from abstracts. The functional impact of phosphorylation on the substrate is critical for the understanding of protein networks and the prediction of the functional outcomes. In this future, we wish to extend the system to cover the impact on other protein properties such as substrate's activation or down-regulation, impact on substrate biological processes and pathways, and protein structure.

# Chapter 5

# IDENTIFYING COMPARATIVE STRUCTURES IN BIOMEDICAL TEXT

## 5.1 Introduction

In Chapter 3, we described miRiaD [23], a tool to capture the role of microRNAs in disease. As indicated earlier, it is important to capture the differential expression of microRNAs in disease samples, even though such statements might not necessarily describe the microRNA's role in the disease. These differential expression statements are important to capture as they can an guide disease diagnosis, assess prognosis, or predict response to a therapy. Thus, to complement miRiaD, we developed a tool called DEXTER [24], which extracts text evidence of genes/microRNA differential expression level in diseases. Our initial analysis of sentences about differential expression information in diseases suggested that a large portion of such statements are stated using **comparative statements** in text, comparing expression levels in two different samples as in sentence (1) below. In this example, the "expression of mir-21" is being compared in two samples ("lung cancer" vs. "normal tissues"). Thus, the development of a system to extract comparison relations from text was critical for the development of DEXTER.

Example 1: The expression of miR-21 was lower in lung cancer tissues compared with adjacent noncancerous tissues.

Additionally, **Association** relations, one of our CAIR relations can be inferred from such comparative relations. Biomedical researchers conduct experiments to validate their hypotheses and infer associations between biological concepts and

entities, such as mutation and disease or therapy and outcome. In such studies, researchers make observations under two different scenarios (e.g., disease sample vs. control sample). When the differences between the groups are statistically significant, associations can be inferred. Comparative studies are prevalent in nearly every field of biomedical/clinical research. Thus, the development of automated techniques to identify such statements would be highly useful even beyond its application for identifying differential expression of miRNAs/genes in diseases.

Comparative sentences typically contain two (or more) entities, which are being compared with respect to some common aspect as in Example 1 above, which compares miR expression level in cancerous vs. non-cancerous tissues: Typically, the entities, which we will refer to as **compared entities**, are of the same type. In the example, the entities being compared are two tissues: "lung cancer tissues" and "adjacent noncancerous tissues". The **compared aspect** ("Expression of miR-21" in this sentence) is the aspect on which comparison between the two entities is being made. The word "lower" indicates the **scale** of the comparison, thereby providing an ordering of the compared entities. These definitions are similar to those described in [114].

In this chapter, we describe a system to automatically identify comparative structures from text [25]. We have developed patterns based on our EDG-based relation extraction framework to identify comparison sentences and also extract the various components (compared aspect, compared entities, and scale). The developed system identifies explicit comparative structures at the sentence level, where all the components of the comparison are present in the sentence. Note the development of this comparison RE system is independent of any downstream text-mining

applications similar to the development of CAIR relations. We will demonstrate the applicability of the comparison RE system in the next chapter, where we describe the text-mining tool DEXTER to extract differential expression level information of genes and microRNAs in diseases. To the best of our knowledge, ours is the only work in the biomedical field that attempts to cover a wide range of comparisons, capture all comparison components (compared aspect, entities, and scale of the comparison), and does not impose any restrictions on the type of compared entities. In the rest of this chapter, we will define the task, describe our approach and comparison patterns, and present the results of our evaluation. We achieved an F-score of 0.87 for identifying comparison sentences and 0.78, 0.81, 0.77 for extracting the compared aspect, scale indicator, and compared entities, respectively.

## 5.2 Related Works

The sentence constructions used to make comparisons in English are complex and variable. Bresman [115] discussed the syntax of comparative clause construction in English and noted its syntactic complexity, 'exhibiting a variety of grammatical processes'. Friedman [116] reported a general treatment of comparative structures based on basic linguistic principles and noted that automatically identifying them is computationally difficult. In [117], the authors proposed a model of comparative interpretation that abstracts from textual variations using descriptive logic representation.

The above studies provide an analysis of comparative sentences from a linguistic point of view. Computational systems for identifying comparisons have also been developed. Jindal and Liu [118] proposed a machine learning approach to classify sentences (comparative vs. non-comparative) from text. The authors extended

their work in [119] to extract comparative relations i.e. the compared entities and their features, and comparison keywords from the identified comparison sentences. In [120], the authors described a machine learning approach to extract and visualize comparative relations between products from Amazon customer reviews. They describe a comparative relation as a 4-tuple, containing the two compared products, the compared aspect, and a comparison direction (better, worse, same). In [121], the authors focused on mining opinions from comparative sentences from product review sentences and extracting the preferred product.

Relatively few works on identifying comparative sentences and/or their components from biomedical text have been developed. Park and Blake [114] reported a machine learning approach to identify comparative claims automatically from full-text scientific articles. They introduced a set of semantic and syntactic features for classifications using three different classifiers: Naive Bayes (NB), a Support Vector Machine (SVM), and a Bayesian network (BN). The focus of this work was on identifying comparison sentences and the extraction of their components was not addressed. Fiszman et al. [122] described a technique to identify comparative constructions in MEDLINE citations using under-specified semantic interpretation. The authors used textual patterns combined with semantic predications extracted from the semantic processor SemRep [123,124]. Their system extracts the compared entities (limited to drugs) and the scale of the comparison. To the best of our knowledge, [122] is the only reported work that goes beyond the identification of comparison sentences to identify the different components of the comparison in biomedical text. But unlike our work, theirs is limited to comparisons between drugs, does not extract the

comparison aspect, and appears to be limited in their coverage of comparison structures.

## 5.3 Methods

### 5.3.1 Task Definition

Basic comparison sentences as indicated earlier contain two or **more compared entities (CE)** and a **comparison aspect (CA)** on which compared entities are being compared. Additionally, there are two parts in such sentences indicating the comparison. The first is the presence of a word that indicates the scale of the comparison and the other separates the two compared entities. The former is often comparative adjectives or adverbs (such as "higher", "lower", "better", etc.), while the latter can be expressed with phrases or words (such as "than", "compared with", "versus" etc.). We will refer to the former comparative word indicating the scale as the **Scale Indicator (SI)** and the latter, separating the entities, as the **Entity Separator (ES).** In example (2) below, the key parts of such a comparison structure are highlighted, which will be extracted from our system.

Example 2: [Arteriolar sclerosis]$_{\mathbf{CA}}$ was significantly [higher]$_{\mathbf{SI}}$ in [addicts]$_{\mathbf{CE}}$ [than]$_{\mathbf{ES}}$ [controls]$_{\mathbf{CE}}$.

In [119], the authors categorized comparative structures into four classes: (1) Non-Equal Gradable, (2) Equative, (3) Superlative, and (4) Non-Gradable. **Non-Equal Gradable** comparison indicates relations of the type greater or less than, providing an ordering of the compared entities as exemplified in Example 2 above. **Equative**

structures indicate an equal relationship between the two entities with respect to the aspect as exemplified in Example 3 below.

Example 3: Candesartan is as effective as lisinopril in reducing blood pressure.

Comparisons, where one entity is "better" than all other entities indicated through words such as "best", "worst", etc. are termed as **Superlative**. An example of such a superlative case is shown in Example 4 below. Sentences in which the compared entities are not explicitly graded are called **Non-Gradable**. In biomedical literature, these non-gradable comparison sentences typically occur in the Methods section of an abstract, where the author is describing the experiment being conducted without indicating the result as in Example 5 below.

Example 4: Patients with hepatoblastoma had the worst outcome of the group.

Example 5: We compared lesion growth between placebo and tissue plasminogen activator-treated patients

In this work, we will be addressing only the first two types: **Non-Equal Gradable** and **Equative** comparison since an explicit comparison and presence of compared entities in the comparison sentence are important of the text-mining tools we intend to develop using this comparison system. Additionally, since we consider processing at the sentence-level only, comparisons where a larger body of text (multiple sentences) contains information about all the components of a comparison, they are not considered in this work. Thus, superlative cases will not be considered because all the compared entities are rarely mentioned within a single sentence and must be inferred from the context.

As indicated earlier, non-gradable comparison sentences are typically "study" sentences in the Methods sections of abstracts, which only defines the experiment and does not present the results of the comparative study. Since the scale of the comparison is absent in such sentences (as in Example 5), we Non-Gradable comparisons are not considered in this work. Typically, the result of such a comparative study in these cases will be indicated in the Results section with another comparison sentence, which would be gradable and thereby captured by the tool.

### 5.3.2 Approach

As discussed earlier in subsection 5.3.1, the two key parts in a basic comparison sentence are a Scale Indicator (SI), indicating the scale of the comparison, and an Entity Separator (ES), separating the compared entities. We will use syntactic dependencies from these SI and ES words to extract the compared aspect and the compared entities. Thus, we will use our EDG-based relation extraction framework (described in Chapter 2) to detect comparison sentences and extract their arguments. We defined rules using the Semgrex [50] template (described in Appendix A), which allows us to define lexico-syntactic patterns based on the lemma, part-of-speech, and dependency edges on syntactic dependencies. This approach of defining lexico-patterns is the same as those used in extracting CAIR relations. Below we will discuss the different syntactic structures to identify Non-Equal Gradable and Equative comparisons. A full list of the developed Semgrex lexico-syntactic rules can be found in Appendix D.

Each Semgrex rule/pattern identifies all components of the comparison, specifically the head of the comparison aspect, entities, and scale. Since the components are typically Noun Phrases (NPs), we look at the outgoing edges from the

head nouns to obtain the NPs corresponding to the comparison components. In the next subsection, we will discuss the development of different comparison patterns.

### 5.3.3 Comparative Patterns

As discussed earlier in subsection 5.3.1, the two key parts in a basic comparison sentence are a Scale Indicator (SI), indicating the scale of the comparison, and an Entity Separator (ES), separating the compared entities. We will use dependencies from these SI and ES words to extract the compared aspect and the compared entities. We have developed rules based on syntactic dependencies for various combinations of the two keys parts. We broadly categorize our comparison patterns based on the Scale Indicator word indicating either Non-Equal Gradable or Equative Comparison.

#### 5.3.3.1 Non-Equal Gradable

Non-Equal Gradable comparison indicates a difference between the compared entities. Based on three part-of-speech tags (POS) of the Scale Indicator, different syntactic structures are possible, as described below. Note that in all the figures depicting the dependency graph the compared aspect is highlighted in blue and the compared entities in yellow.

***Comparative Adjective:*** Starting with the most frequent case for Scale Indicator, which is a comparative adjective(JJR) such as “better”, “higher”, “lower” etc., there are two broad categories of syntactic structures which we consider. The **first** category involves copular structures, where the JJR serves as the predicate of the comparison relation. The compared aspect is typically the subject of the JJR as shown in Figure 5.1. Thus we follow the *nsubj* edge from the JJR to get the head of compared

aspect. We use the *nmod:than* from JJR to extract one of the compared entities. The second entity will also have an edge from the JJR, which can be prepositional edge (nmod:in as in Figure 5.1). Thus we use *nmod* edges from the predicate JJR to determine the second compared entity. Note all prepositional edges such as "with", "for", "during" etc. are considered. Additionally, the second compared entity will be separated by an Entity Separator ("than" in this case) from the first compared entity. Thus we further verify that the extracted compared entities are separated by an ES.

The position of the entity separator "than" is critical for determining the second compared entity as well as the first compared entity. As shown in Figure 5.2, despite the similar copular structure to the sentence in Figure 5.1, the subject of the JJR ("better" in this case) is the compared entity rather than the aspect. This is because the JJR is followed by the ES "than". Thus ordering of the words is an important clue when differentiating between these cases.

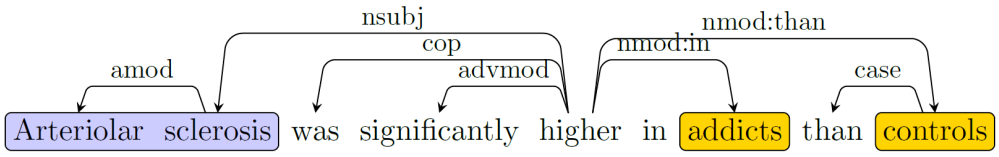


Figure 5.1 Comparative Adjective copular form 1

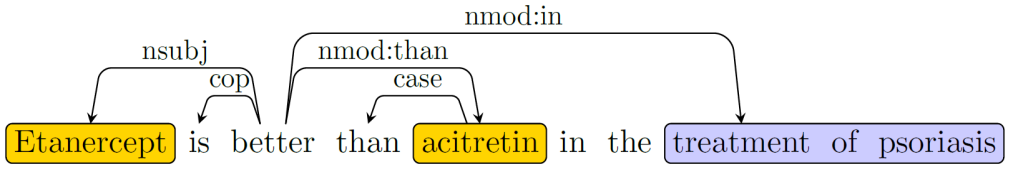


Figure 5.2: Comparative Adjective copular form 2

The **second** category involves sentences, where the comparative adjective modifies a head noun and this modified noun provides the compared aspect, as shown in Figures 5.3 and 6.4. Since the compared aspect is modified by the JJR, we used the *amod* edge to detect the aspect. In these cases, the noun phrase containing the Scale Indicator will be connected to a verb and typically serves as the predicate of the comparison relation. The entity separator in the sentence in Figure 5.3 is "compared to" and we can extract one of the compared entities ("intravenous morphine") by following the *advcl:compared_to* edge from the predicate verb ("offers").

Note that in the first example (Figure 5.3), the Verb Group ("offer") is in the active form, and in the second example (Figure 5.4), it is in the passive form ("was observed in"). Due to the active/passive form difference, the aspect is in the object position and one of the compared entities is in the subject position in the first example, while the reverse is true for the second example. In the dependency representation, the *nsubj* edge and the *nmod:in* edge provide the subjects in active and passive cases and *dobj* and *nsubjpass* provide the possible objects. Note that in certain cases, the author might use an adjective (JJ) instead of the comparative form ("high" instead of "higher"). We treat such cases in the same way we treat the comparative adjective (JJR) form.

Note that the Semgrex patterns only identifies the head words of the various components, which are typically NPs. We follow outgoing dependency edges from these head words to extract phrases corresponding to each comparison component. For example, in Figure 5.1 "sclerosis" is identified as the aspect head and we follow the edge *amod* to extract the aspect phrase "Arteriolar sclerosis". In Figure 5.4, we extract

"TP expression" as the aspect phrase and not "Higher TP expression" as "higher" is the trigger of the comparison and identified as the scale.

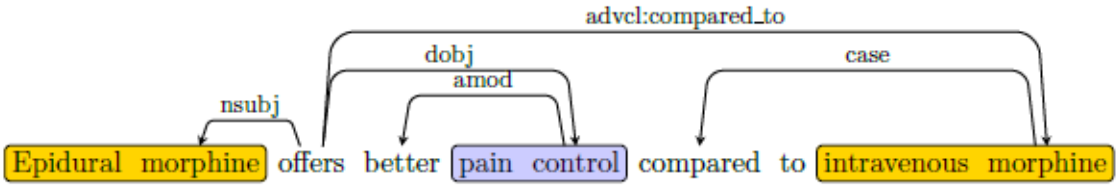


Figure 5.3: Comparative Adjective modifier form 1

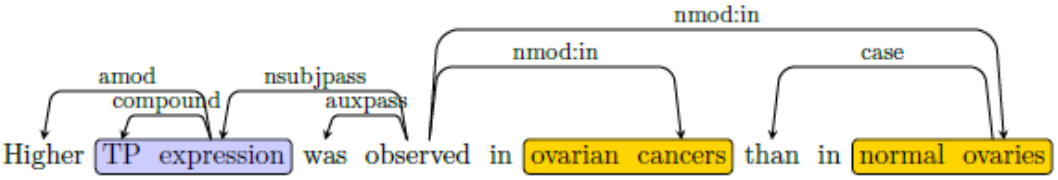


Figure 5.4: Comparative Adjective modifier form 2

***Comparative Adverb***: In these sentences, the comparison scale is indicated through comparative adverbs (RBR) such as "more", "less" etc.. Typically, the RBR modifies an adjective (JJ) as shown in Figure 5.5, where the adjective is "effective".

This adjective serves as the predicate of the comparison and dependency edges from it are used to determine the aspect and entities. The syntactic structure and our rules are very similar to the first category of the Comparative Adjective case. Thus we use the *nsubj* and *advcl:compared_*to edges from "effective" to determine the compared entities. Note that the compared aspect in this example is a clause headed by

a VBG ("reducing MCP-1 levels") and thus in addition to *nmod* edges, we need to consider the adverbial clause modifier (*advcl*) edge to determine the aspect.

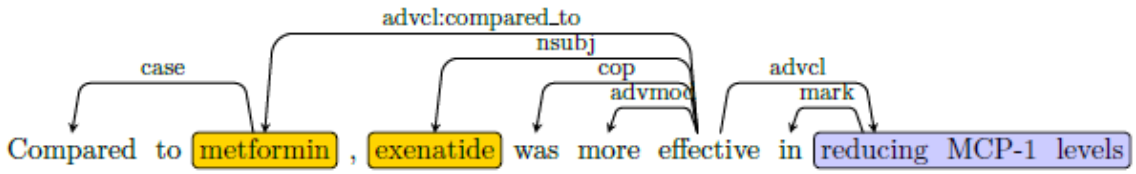


Figure 5.5: Comparative Adverb form

***Verbs:*** Certain verbs such as "increased", "decreased" as well as "improved" indicate differences and can be used as a SI. This verb serves as the predicate of the comparison relation and outgoing dependencies can be used to determine the arguments of the comparison. We have observed two categories based on the voice (passive vs. active) of the Verb Group containing this verb. The passive case is depicted in Figure 5.6a ("was increased in"). In this case, we follow the *nsubjpass* edge to determine the compared aspect. In Figure 5.6b, since the scale indicator "improved" is in the active voice, the direct object of the verb will instead provide the aspect. Extraction and verification of the compared entities is similar to the cases described previously (e.g. *nmod:in* in Figure 5.6a; *dobj* and a*dvcl:compared_with* in Figure 5.6b).

Note that a verb in past participle tense (VBN) can be used as an adjective and modify a noun (e.g., Increased TP expression was found in ...). We treat cases when the scale indicator verb is used as a modifier of an NP like the second category of Comparative Adjectives.

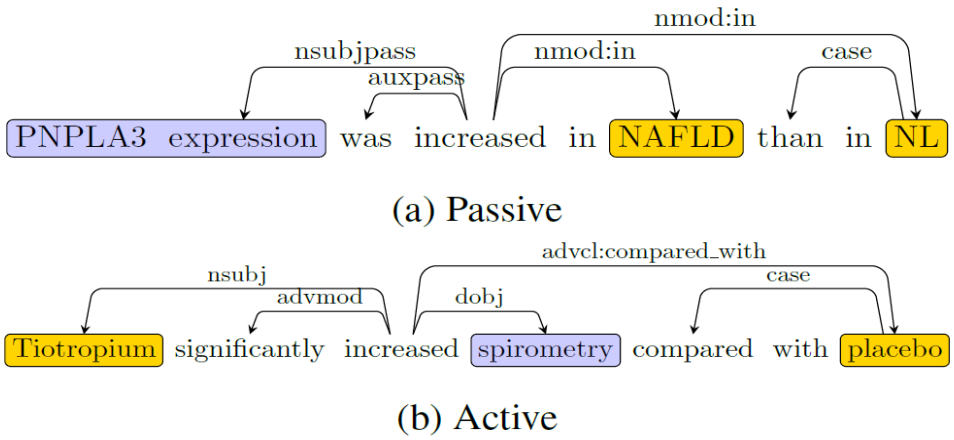


Figure 5.6: Comparative verb form

#### 5.3.3.2 Equative

A sentence with Equative comparison corresponds to cases where the result of comparison indicates no difference between the compared entities (as in Figure 5.7). In these cases, it is very rare to find the usual Entity Separator (ES) and instead words such as conjunctions ("and", "or"), "between" and "among" play the role of the ES. We have observed three frequently occurring types of such Equative comparative structures.

The **first** category involves the structure "X as JJ as Y", where JJ is an adjective. In these cases, the adjective serves as the predicate of the comparison. Figure 5.7 depicts such a case, where the adjective is "effective". Here one of the compared entity "botox" is the subject of the JJ "effective". The second compared entity "oral medication" is preceded by the ES "as" and a *nmod:as* edge from the JJ to the entity is present. The compared aspect is typically attached to the second compared entity through a *nmod* edge (*nmod:for* in this case). Note that the ES "as" need not appear immediately after the JJ (e.g. "Botox is as effective for overactive bladder as

oral medication"). Due to the "CCProcessed" (discussed in Chapter 2) representation [125] of collapsing edges we can still consider the *nmod:as* from "effective" to determine the second compared entity. The only difference, in this case, is that the *nmod:for* edge used to determine the aspect is from the predicate "effective".

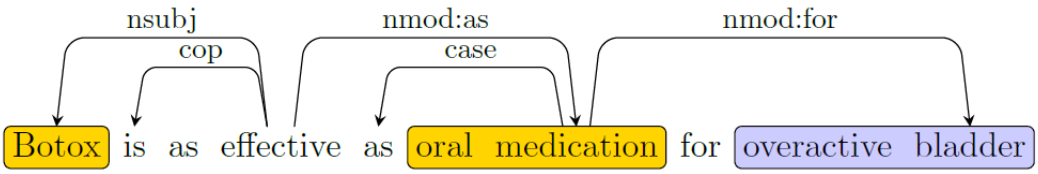


Figure 5.7 Equative Form 1

The **second** case involves the Scale Indicator phrase "similar to" as shown in Figure 5.8. Here the subject of the adjective "similar" is the compared aspect. The *nmod* edges (*nmod:in* in this example) from "similar" are used to determine the compared entities. The entities in these cases are separated through conjunctions. Note that the SI "similar" can also modify the compared aspect (e.g. "Similar CA was observed in CE1 and CE2"). This case closely resembles the second category of comparative adjectives and similar rules are used.

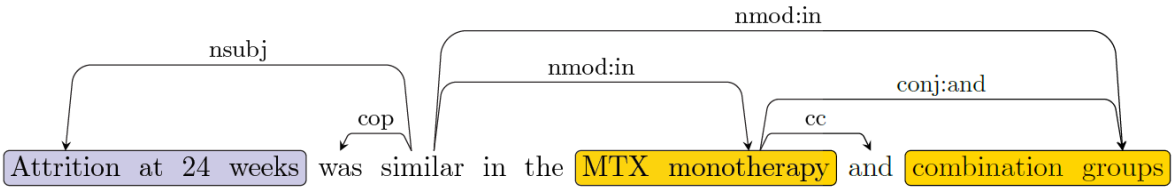


Figure 5.8 Equative Form 2

The **third** category involves Scale Indicator phrases such ‘‘no differences”, “no changes” etc. Similar to the case of the second category comparative adjectives, here the SI “difference” is part of a NP and hence is connected to a verb, which serves as the predicate. Typically these verbs can be “linking” verbs (‘is”, “was” etc.) in active form or certain verbs indicating the presence (“found in”, “noted in”, “observed in”) in the passive form. In active voice case, as shown in Figure 5.9, the SI typically follows an existential such as “there”. In these cases, the *nmod:between* from the predicate verb ( “was” in this case) is used to determine the compared entities. Other *nmod* edges we consider are *nmod:among* and *nmod:in*. The compared aspect is attached to the second compared entity though *nmod* edges (*nmod:for* in this example). A large proportion of Equative structures do not mention the compared entities explicitly, and as per the definition of our task, we do not extract the comparison components in these cases.

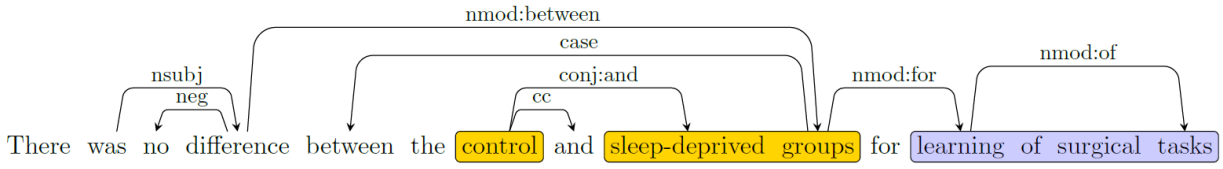


Figure 5.9: Equative Form 3

## 5.4 Evaluation

We evaluated our system for its effectiveness in identifying comparative sentences and its components on a test set of 189 comparisons from 125 abstracts annotated by an annotator, who was not involved in the design and development of the system. In the next subsections, we will describe the annotation methodology, present

the results, and provide an analysis of errors. Note, in the next chapter, we will describe DEXTER, a tool to extract differential expression information, which is a direct application of this comparison RE system. We conducted a separate evaluation for DEXTER, details of which are discussed in the next chapter.

### 5.4.1 Experimental Setup

To evaluate our system's performance, we first created a test set of 125 abstracts. We selected abstracts that usually draw conclusions by comparing two contrasting situations. Randomized controlled trials (RCT), which compare the outcome between two randomly selected groups, fit this definition well. For this reason, we searched for RCTs in PubMed with the query "(Randomized Controlled Trial [Publication Type])". This query yielded 431,226 abstracts. However, since the RCT set of abstracts are about medical/clinical trials, we felt we needed to include abstracts that also discuss comparisons of aspects involving biomolecular entities. Thus, we considered a second set of abstracts related to diseases and genes. As we target to identify comparison sentences, we chose abstracts tagged as "comparative study" in PubMed because they tend to contain comparisons. Then to choose abstracts about comparisons involving biomolecular entities, we used the PubMed query: "(Comparative Study [Publication Type]) AND expression[TIAB] AND (cancer[TI] OR carcinoma[TI])", restricting the comparative studies to gene expressions and cancer-related studies. This query yielded 8,479 abstracts.

We combined the two sets of abstracts and chose a subset of abstracts that included 100 abstracts from the RCT set and 25 from the second set. These 125 abstracts were then given for annotation by a biomedical researcher expert who did not take part in the development of the system. We asked the annotator to annotate all

sentences in the 125 abstracts that indicated an explicit comparison i.e. indicated the scale of comparison and thus a gradable comparison and not a mention of a planned investigation (Non-gradable).150 sentences from the 125 abstracts were annotated as comparison sentences and included 189 comparison relations. Additionally, for each comparison relation, we asked the annotator to mark the four components of the comparison: the compared aspect (CA), the two compared entities (CE1 and CE2), and a word or phrase that indicates the scale of comparison (SI).

### 5.4.2 Results and Discussion

Annotations of the test set of 125 abstracts yielded 189 comparisons, each containing a compared aspect, a scale indicator, and two compared entities. We ran our system on the test set and evaluated its performance on correctly identifying the (1) comparison sentences, (2) compared aspect (CA), (3) scale indicator (SI), and (4) compared entities (CE). When computing true positives (TP), we compared the head word of the annotated components (CA, SI, CE) with the head words extracted by our system. A mismatch resulted in both a false positive (FP) and a false negative (FN). Note, since there can be two compared entities (CEs) in an annotated comparison relation, if one of the CEs is not detected by our system, it resulted in a FN. If both CEs were not detected by our system, it resulted in two FNs. Components (CA, SI, CE) extracted by the system from a sentence in an abstract not annotated as a comparison sentence resulted in FPs. We computed Precision (P), Recall (R), and F-score (F) measures for each evaluation type, results of which are shown in Table 5.1.

| **Type** | **Precision** | **Recall** | **F-Score** |
|---|---|---|---|

| | | | |
|---|---|---|---|
| **Sentence** | 0.91 | 0.83 | 0.87 |
| **Comparison Aspect** | 0.85 | 0.72 | 0.78 |
| **Scale Indicator** | 0.87 | 0.75 | 0.81 |
| **Compared Entities** | 0.84 | 0.72 | 0.77 |

Table 5.1: Evaluation Results for Comparison

We analyzed the errors made by our system and the majority of the errors (more than 80%) encountered were due to incorrect parsing of complicated sentences. For example, in sentence (3), the clause modifier edge *acl* to "compared" was from "feed" instead of the "palatable", the annotated aspect. If the parse had been correct, then our EDG rule would have correctly extracted the comparison.

Example 3: Pro-Dynam was significantly less palatable, with significantly less consumption of treated feed compared with either Equipalazone Powder or Danilon Equidos.

Another set of errors involved cases such as the sentence in (4). The issue here is that the scale indicator ("superior") was tagged as an adjective (JJ) and not a comparative adjective (JJR). Since our treatment of such patterns was limited to JJR scale indicators, we missed these cases.

Example 4: Gatifloxacin is superior to moxifloxacin in shorter treatment regimens for multidrug-resistant TB

Recall, Fiszman et al. [122] is the only reported work that goes beyond the identification of comparison sentences to identify the different components of the comparison in biomedical text. Although their work attempts to tackle the similar task of identifying comparison sentences and its components, we do not directly compare

with their results. This is due to the fact that their implementation is limited to "direct comparisons of the pharmacological actions of two drugs". We ran their system on our annotated test data and only 8 out of the 189 comparisons were identified by their system as their implementation only detects comparison if the two compared entities (CEs) are drugs. We also ran their system on some artificially created sentences obtained by replacing CEs with drugs and observed that their system seemed limited in the coverage of comparison structures.

## 5.5 Conclusion

In this chapter, we have presented a system to identify comparison sentences and extract their components from literature. We have applied this comparison RE system to develop a tool called DEXTER that extracts differential expression level information of genes/microRNAs in disease samples. Even though the initial motivation behind the development of this system was for the development of DEXTER, the significance of such a system arises from the prevalent nature of comparative structures in the biomedical literature. We have observed that in a sample of abstracts describing randomized controlled trials or comparative studies, almost every abstract contained at least one comparison. Moreover, other text-mining applications might rely on extracting the arguments of a comparison. Additionally, biomedical researchers conduct experiments to validate their hypotheses and infer associations between biological concepts and entities. The results of these experiments are typically stated through comparison sentences in literature and thus can be used to infer Association relations, one of our CAIR relations. To the best of our knowledge, ours is the only work that attempts to cover a wide range of comparisons, capture all

comparison components, and does not impose any restrictions on the type of compared entities.

# Chapter 6

# EXTRACTING EXPRESSION IN DISEASE

## 6.1 Introduction

microRNAs (miRs) post-transcriptionally regulate the expression of their target genes, and abnormalities in miR expression have been associated with many diseases [84,126–130]. Expression levels of certain genes and miRs can guide disease diagnosis, assess prognosis, or predict response to therapy. Identifying them is a key aspect of precision medicine [131]. Thus, the development of text-mining systems to connect miRs to diseases is a major part of this dissertation research. In chapter 3, we described miRiaD [23], a tool to extract the role of miRs in diseases. We had also indicated that expression level information of miRs in disease samples do not necessarily convey the role of miR in disease but is important to capture as it is another aspect of connecting a miR to a disease.

In this context, we developed an automated text-mining tool, DEXTER (Disease-Expression Relation Extraction from Text) [24] to extract information on miR's differential expression level in a disease sample compared to a non-diseased sample (e.g. cancer tissue vs. normal tissue). Since miR expression regulate target gene expression and abnormalities in gene expression level in disease tissues/cells is equally useful for biologist and researchers, DEXTER also considers expression information of genes. Specifically, DEXTER extracts the expressed gene or miR, the

associated disease, the expression level (e.g., high or low), and samples where the expression level is compared.

Note, these differential expression level statements are typically stated through comparison sentences, where the expression level of a gene/miR is compared under two different samples as in Example 1a. Thus DEXTER is a direct application of the comparison relation extraction system that we discussed in Chapter 5. Additionally, there are certain expression in disease statements, where the expression level in disease are not explicitly compared to another sample as in Example 1b but are important to capture. We have developed a new relation called "found-in" using our EDG-based RE framework to extract such expression level statements.

Example 1a: The expression of miR-21 was **lower** in lung cancer tissues **compared with** adjacent noncancerous tissues.

Example 1b. miR-95 was over-expressed **in** human prostate cancer specimens.

In Section 6.2, we first discuss existing expression databases and motivate the need for DEXTER. Next, in Section 6.3 we formally describe the task and the different types of information that we extract. In Section 6.4 we present our details on the different aspects of developing DEXTER and in Section 6.5 we discuss the results of running DEXTER on a large set of PubMed abstracts. Finally, in Section 6.6 we provide a comprehensive evaluation of the system.

## 6.2 Existing Expression Databases

Most of the existing expression related databases are for gene expression, which are discussed here except miR2Disease [20], miRCancer [21], and dbDEMC [92,93] (already discussed in chapter 3). The development of microarray and next-generation sequencing technologies has led to an abundance of transcriptome-wide gene expression data. Much of this data is publicly available through general repositories such as Gene Expression Omnibus (GEO) [132] and Array Express [133], as well as through more specialized resources, such as International Cancer Genome Consortium (ICGC) [30] and the Cancer Genome Atlas [29] (TCGA: http://cancergenome.nih.gov/), which focus on cancer data, and Tissue-specific Gene Expression and Regulation (TiGER) [134], which organizes gene expression data by tissue type. High throughput mass-spectrometry (MS) is providing expression data at the protein level. This data is captured in resources such as dbDEPC [135,136] a database containing over 4000 differentially expressed proteins in 20 cancers, obtained from 331 MS experiments.

The scientific literature is a rich source of information on specific gene expression-disease relationships that have been observed in thousands of small-scale studies. In general, these results are only accessible through laborious manual curation; however, automated text mining tools are beginning to lower the barriers to systematically capturing this data. Several resources focus on manually curated data from publications on disease-related gene and miRNA expression such as DisGeNET [137,138], OncomiRDB [139], and miRCancer [21].

Finally, the BioXpress [28,140] database was developed to address the need for an integrated view of cancer gene and miRNA expression data from a variety of studies, both large and small-scale. BioXpress collects expression data from publicly

available sources such as TCGA [29], and ICGC [141] and uses a standardized statistical method to identify the significance of differential expression of genes and microRNAs between tumor and adjacent non-tumor samples from the same patient. In addition, BioXpress reports differential expression of genes manually extracted and curated from publications and supplemental information, which enables researchers and clinicians to easily compare patients' expression data with existing knowledge from literature. While there is substantial expression information obtained from large-scale studies in BioXpress (18626 genes and 710 microRNAs from 33 cancer types and 667 patients), manually curated annotations based on information from the literature (138 genes-PMID annotations) lag significantly. Incorporation of automated text mining tools, like DEXTER, has the potential to streamline and accelerate the BioXpress curation process. One of the motivations in developing DEXTER was to extend the BioXpress database.

## 6.3 Approach

### 6.3.1 Types of Expression Information

As indicated earlier, among the sentences in the literature regarding gene expression in disease, we have observed two broad categories:

**Type A:** In this category, sentences compare the expression level of gene/miR in two differing scenarios, at least one of which involves a disease. For instance, in Example 2a below, the expression of Shp2 is compared between two types of tissues, one of which is cancerous (OSCC).

Example 2a: **Expression of Shp2** protein was significantly <u>upregulated</u> <u>in</u> **OSCC tissues** compared with the **normal tissues**

Note that in this example, the compared groups are cancerous and normal tissues; these are the sentences of interest to BioXpress. They also allow us to infer an association between the miR/gene expression level and the disease. However, not all Type A need to contrast expression levels between disease samples and normal samples. In Example 2b below, the contrasted scenarios are metastatic and primary tumors instead.

Example 2b: Higher **miR-210 expression** was found in **metastatic tumors** compared to **primary tumors**

**Type B:** In the second category are sentences that indicate the expression level of a gene/miR in a disease state, but without an explicit comparison. In Example 3, the expression of miR-155 is reported to be over-expressed (high expression) in a disease sample ("pancreatic cancer tissues") without any explicit comparison to another sample. Here, the comparison of the expression level of a gene/miR between a disease and non-disease sample may be implicit.

Example 3: **miR155** was overexpressed in **pancreatic cancer** tissues.

Note, there are certain expression level information sentences, that state the connection between a gene/miR's expression level and various disease-related concepts such as disease outcomes (e.g., "poor survival") or disease processes (e.g., "metastasis", "cancer cell proliferation"). Examples 4a, 4b represents such cases. While such sentences are frequently found in the literature and inform us about the impact of a gene's expression (high or low), it is not clear whether the gene/miR was expressed *naturally* in the diseased tissue or cell. For instance, from Example 4a, we

do not know whether C1GALT1 over-expression is typically observed in breast cancer; all we know is that *when* C1GALT1 is over-expressed in breast cancer, cell growth, migration, and invasion are enhanced. Moreover, it is possible in these cases that the gene's expression is being experimentally manipulated and is not a natural property of the disease cells at all. Therefore, we do not extract information of a gene/miR expression level in disease from such sentences for purpose of this tool. Additionally, this association information between the miR's expression and disease concept (disease process, outcome) as in Example 4b will be extracted through our miRiaD tool (described in Chapter 3) through CAIR relation.

Example 4a: Overexpression of C1GALT1 enhanced breast cancer cell growth, migration, and invasion in_vitro as well as tumor growth in_vivo.

Example 4b: high mir21 expression is associated with lung cancer metastasis.

### 6.3.2 Task Definition

Based on the discussion above, we focused on information extraction from Type A and Type B sentences. For both types, DEXTER extracts the **expressed gene/microRNA**, the **expression level**, and the **associated disease.** For Type A sentences, where the expression is contrasted under two scenarios, it also extracts the **compared scenarios.**

To summarize, given a text, our tool, DEXTER, extracts:

a) **Expressed Gene/microRNA:** The differentially expressed gene (normalized to NCBI Gene ID [142]) / microRNA.

b) **Associated Disease:** The disease associated with the sample where the gene is expressed. The disease is normalized to a Disease Ontology [143] ID (DOID).

c) **Expression Level:** The level of expression, normalized to either "High" or "Low".

d) **Disease Sample:** the sample (e.g., tissue, cell, cell line, etc.) mentioned in the sentence, where the gene is expressed.

e) **Compared Sample**: A second sample, which is used as a contrast to the sample in (d). This information is available in Type A, but not Type B, sentences.

Consider the sentence in Example 2a. From this sentence we will extract the following:

(a) Shp2 (NCBI Gene ID: 5781), (b) OSCC (Oral Squamous Cell Carcinoma; Disease Ontology DOID:0050866), (c) upregulated (High), (d) OSCC tissues, (e) normal tissues

Recall, one of the motivations behind DEXTER was to extend the literature portion of the BioXpress [28] database. However, for BioXpress additional constraints need to be considered as BioXpress reports only differential expression of genes and microRNAs between cancer and normal (non-tumor) samples. Thus, given BioXpress criteria, appropriate DEXTER's extractions will be flagged to be included in the literature-based portion of the database. For example, in sentence (2a) is relevant for BioXpress since the expression level in cancerous tissues and normal tissues are being compared. Information extracted from Type A sentences describing other compared scenarios (e.g., in Example 2b or 5 below) and not flagged for inclusion to BioXpress but are saved as they are of potential interest to researchers, clinicians, and curators of other disease resources such as dbDEMC [92,93].

Example 5: **expression levels of miR-454-3p** were <u>higher in</u> **high grade gliomas** than in **low grade gliomas**. [PMID: 25190548]

## 6.4 Methods

The first step in developing DEXTER is the extraction of relations that correspond to Type A and Type B sentences. We will use our EDG-based RE framework to extract such relations, which will be discussed in Section 6.4.1 and 6.4.2. Next, we detect and tag all relevant entities and phrases (gene, miR, disease, expression phrase), details of which will be discussed in Section 6.4.3. Finally, we will determine whether the arguments of relations extracted are of the correct type for DEXTER and extract final relation that can be put into a database. Details of argument filtering and extraction will be discussed in Section 6.4.5.

### 6.4.1 Relations for Type A: Comparison Constructions

Recall that expression in disease samples for Type A information are present in comparative sentences, where the expression of a gene/microRNA is compared under two or more scenarios. In chapter 5, we have discussed a system to detect comparative sentences from text and also extract its arguments: (1) **Compared Aspect (CA)**: the quantity being compared, (2) **Compared Entities (CE1 and CE2):** the scenarios being compared, and (3) **Scale Indicator (SI)**: the scale of the comparison. Consider the sentence in Example 6 below.

Example 6: Plasma miR-187 was significantly higher in OSCC patients than in normal individuals.

From this sentence, our comparative relation extraction tool (discussed in chapter 5), will extract “Plasma miR-187” as CA, “OSCC patients” and “normal individuals” as CE1 and CE2 respectively, and “higher” as the SI.

### 6.4.2 Relations for Type B

Type B sentences indicate the expression level of an entity (e.g., miR) in some disease sample, without explicitly contrasting it with another state. Importantly, an expression level for the entity, not just the entity itself, is mentioned. Hence we are interested in the (1) **Expressed Aspect (EA)**: the entity being expressed, (2**)** **Expressed Location (EL):** the biological context of the expressed entity, which can be disease samples, cells, tissues, etc. and (3) **Level Indicator (LI)**: a phrase indicating the level of expression.

We have developed relation called "found-in" described in the next section using our RE framework, which extracts the presence/absence of entities (genes, miRs) in anatomical parts, tissues, etc. We will use the *arg0* argument of found-in as EA and *arg1* as EL. For extracting LI, we need to consider two scenarios with respect to the trigger of the "found-in" relation. The first case involves triggers such as "overexpressed in", "under-expressed in", "upregulated in", "increased in", where the trigger provides us with the LI, as in Example 7a. For the second class of found-in triggers such as "is found in", "is detected in", "is increased in" etc, the LI modifies (through "nmod:of", "compound", "amod" syntactic edges) the Expressed Aspect as in Example 7b. Thus, we will extract "GALNT2", "oral squamous cell carcinoma" and "overexpressed" as EA, EL, and LI respectively for sentence 7a, and "mir-155", gallbladder cancer" and "high" as EA, EL, and LI respectively for sentence 7b. A complete list of words/phrases corresponding to different found-in and LI triggers can be found in Appendix C.2 and C.10 respectively.

Example 7a: GALNT2 is frequently <u>overexpressed in</u> oral squamous cell carcinoma.

Example 7b: High level of mir-155 was <u>found in</u> gallbladder cancer.

#### 6.4.2.1 Extracting Components of Type B: Found-in relations

To extract components of Type B expression relations, we have developed new *arg0/1* semantic relation called "found-in", which indicates the presence/absence of genes/microRNAs in anatomical parts. Anatomical Parts such as cells, cell lines, tissues, and organs, etc. are not directly affected by entities (genes and microRNAs). Authors typically mention the presence/absence of entities in anatomical parts. The knowledge of the presence or absence in a cell can have a significant bearing on the understanding of the biology of the cell and detection of such found-in relation will be useful in different RE systems.

There are two classes of triggers used to detect such relations. First set of triggers include words or multi-word triggers like: "found in", "detected in", "occurred in" etc. The second set of triggers to include: "overexpressed in", "highly expressed in", "upregulated in", which in addition to conveying the Found_in relation also indicates the expression level. Typically, in these cases we follow the *nsubjpass* and *nmod:in* edges to add the arg0 and arg1 edges as in Figure 6.1. Another syntactic structure conveying the Found_in relation is presented in Example 6.2. In these cases, the *arg0* is added using the *dobj* edge from the lexical trigger (here "found"). A full list of the different lexico-syntactic variations can be found in Appendix D.

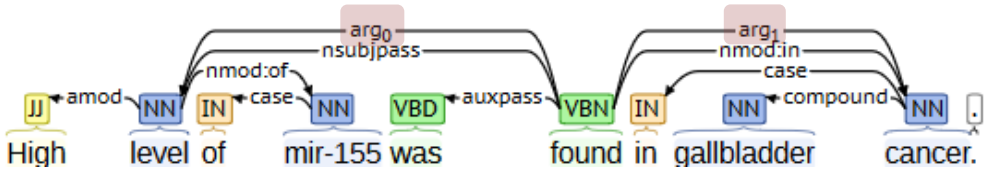


Figure 6.1: Example 1 EDG for Found_in

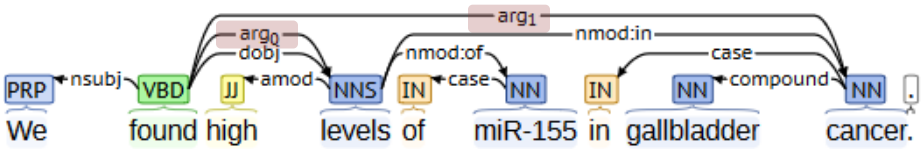


Figure 6.2: Example 2 EDG for Found_in

### 6.4.3 Entity Detection and Phrase Typing

Since we are interested in extracting expression information in disease, the arguments/components of our relations should satisfy certain type constraints. For example, in a comparison construction, the Compared Aspect must of type *gene expression*. Further, our task requires the extraction of the expressed gene/miR, expression level, and the associated disease. Therefore, we need to determine the type of argument phrases. In this phase, we look at noun phrases of the arguments and determine if they contain terms that refer to entities of type *gene/miRNA, expression, or disease/disease-sample*.

Note that in this step we only tag all the genes, microRNAs and diseases, expression, and disease-sample phrases in the text; details about how the particular expressed gene and the associated disease is extracted will be described in the Section 6.4.3. Gene[4] mentions are detected using PubTator [98]. We downloaded and used the pre-computed annotations from PubTator, which contains gene mentions in abstracts normalized to NCBI Gene IDs. For microRNA detection, we use regular expressions as already described in Chapter 3, Section 3.3.2. To determine whether a phrase is of type "*Expression*", we check the head noun of the phrase against a list of expression

[4] We don't distinguish between genes and proteins in this phase

triggers such as “expression”, “level”, “over-expression” etc. Note, the gene/microRNA whose expression is defined by such an *Expression* phrase will either be in the same NP phrase indicating the *Expression* phrase or attached to it through a prepositional phrase. In both cases, the expressed gene will modify the Expression phrase such as in “X expression” or “Expression of X”, where X is a name of a gene or microRNA.

To detect diseases, we also use PubTator [98], where disease mentions are normalized to MEDIC IDs [144]. These IDs are mapped to Disease Ontology IDs (DOIDs) using the table provided by Disease Ontology [143], which maps MEDIC IDs to DOIDs. The choice of normalizing diseases to DOIDs was made to allow easy integration to BioXpress, which only uses DOIDs. Note, that the arguments of our relation can be a disease (as detected by PubTator) or contain a disease with its head word matching certain *disease-sample* triggers such as “tissues”, “cells”, “patients”, “samples”, “tumors” etc. A full list of expression and disease sample triggers can be found in Appendix C.7 and C.8 respectively.

### 6.4.4 Argument Filtering and Extraction

There are two primary steps in this phase: (1) verify if the arguments found by the relation extraction modules meet the type constraints and filter accordingly and (2) extract the final relation that can be put into a database.

In this step, we check that the arguments from the RE module (found-in, comparison) are of the right type. Consider the comparison case: we verify that the phrase identified as the Compared Aspect is of type *expression* and that the two compared entities are of type *disease/disease-sample*. Similarly, we verify the type

constraints for Type B: i.e., that the EA is of type *gene* or *gene-expression* and the EL is of type *disease-sample*.

Next, we discuss how to extract all the relevant information to populate a database. We first discuss the extraction of the gene/miR and level and later discuss the extraction of the disease. The gene and level are always extracted from the sentence whereas the disease might be extracted from some other part of the abstract or its title.

#### 6.4.4.1 Expressed gene/microRNA and Expression Level Extraction

Recall the Compared Aspect and Expressed Aspect arguments of Type A and Type B are noun phrases (NP) of type "*expression*" or sometimes the gene itself. Thus these NPs will either directly contain the name of the gene we are capturing or the gene name will be attached to the *expression* phrase as a modifier. We use the gene/miRNA mentions detected in Entity Detection and Typing Module described in Section 6.4.2 to extract the particular expressed gene/microRNA from the Compared/Expressed Aspect arguments.

In addition to extracting the expressed gene, we also need to note the level of expression (high or low). As stated earlier the phrases, the expression level can be the predicate of the extracted relations (e.g. X *higher* in Y than Z, X *over-expressed* in Y) or attached to the Compared/Expressed Aspect phrases as noun-modifiers (e.g. *Lower* expression of X was found in Y). These phrases are already captured by our relation extraction system as Scale Indicator or Level Indicator arguments for Type A and Type B relations respectively. They are then normalized to High or Low by matching them against a list of triggers. We use triggers such as "over-expressed", "high", "increased" etc. to assign **High** expression level and triggers such as "under-

expressed", "low", "decreased" etc. to assign **Low**. A full list of these triggers is listed in Appendix C.10.

#### 6.4.4.2 Extracting the Disease

In most cases, the disease is mentioned in the noun phrases corresponding to the Compared Entity or Expressed Location arguments of the Type A/B relations or attached to it by a prepositional phrase. Thus, while determining the associated disease, we check if a disease detected by PubTator is mentioned in one of the compared entities or in the expressed location argument. In some cases, the arguments of the relations might only contain generic disease phrases such as "tumor", "cancer", "disease" or population phrases such as "patients", "men" etc. (as in the Compared entities in Example 8 below). In these cases, we assume that the referred disease can be inferred from context and the associated disease is extracted from elsewhere in the same abstract. The approach for inferring the referred disease from context is the same as the methodology discussed in Chapter 3 Section 3.3.3.

> Example 8: Conversely, the expression of miR-143 and -195 in **cancer tissues** was significantly lower compared to that in normal tissues.

#### 6.4.4.3 Determining Compared Sample Type

The compared entities extracted from comparison constructions in Type A sentences should be a disease-sample such as a disease cell, tissue, cell line, tumor, patients, etc. Since BioXpress database guidelines require expression data that includes direct evidence of gene expression differences between tumor and adjacent non-tumor tissues (control), we differentiate between comparison to *Control* and *Not-Control* by adding a **frame-of-reference** flag. If one of the compared entities' noun

phrase contains words such as “control”, “normal”, “healthy”, “adjacent” etc. as a noun modifier, we detect the frame-of-reference as ***Control*** (Example 9a) indicating the differential expression in disease versus normal. If no such phrase is detected in the compared entities, we set the flag to ***Not-Control*** as in Example 9b, where the comparison is between two disease subtypes (“T1 bladder carcinoma” and “Ta carcinomas”). A full list of words/phrases usued to determine Control can be found in Appendix C.9.

Example 9a: Higher TP expression was observed in ovarian cancers than in **normal** ovaries. [PMID: 15628771]

Example 9b: “….the expression of PDECGF in **T1 bladder carcinoma** was twofold higher than that in **Ta carcinomas**.” [PMID: 9070497]

Note differential expression between tumor and normal can also be conveyed through certain predicate triggers of Type B relations such as “over/under-expressed”, “increased”, “decreased”, and “elevated”, reduced”. In addition to indicating high/low expression of the gene in cancer cells, these sentences also suggest an *implicit comparison to control*. The use of predicate “overexpressed” used to detect a high level of expression in the disease state does not make sense unless it is a reference to some baseline. In such cases, we assume the comparison reference is normal (non-disease state) and assign the frame-of-reference flag ***Control_Implicit.*** On the other hand, predicate triggers such as “high”, “low”, etc., indicate expression information but does not necessarily imply differential expression. In such cases, we assign ***None*** as the frame-of-reference as in Example 10.

Example 10: Expression of GCS was <u>high in</u> estrogen receptor ( ER ) - positive and HER-2 negative samples [PMID: 24456584]

## 6.5 Large-scale processing for BioXpress

We processed the entire MEDLINE literature for differential expression information in cancer and text-mined results were made available for curation into BioXpress [28]. DEXTER output is appropriate for BioXpress (i) the extracted disease is a type of cancer (as determined by Disease Ontology [143] ID) and (ii) there is an explicit/implicit comparison of expression in cancer samples to normal samples (the frame-of-reference flag is "Control" or "Control_Implicit").

DEXTER was applied on a large set of PubMed abstracts related to cancer. To select cancer-related abstracts, we used the PubMed query 'cancer OR cancers OR carcinoma OR carcinomas OR neoplasm OR neoplasms', which returned 3,717,745 Next we selected only those abstracts that contain certain expression words/phrases, which reduced the number of abstracts to 1,750,928. We ran DEXTER on these abstracts and extracted differential expression information relevant to be included in BioXpress i.e expression of a gene in cancer compared to normal. This processing resulted in 24,416 unique gene-cancer type pairs, which has been integrated into the literature portion of the BioXpress [28] database.

## 6.6 Evaluation

Recall one of our motivation for designing DEXTER was to assist with the curation of the BioXpress database. Thus our first evaluation focuses on results relevant to BioXpress and thus we only consider cases that compare gene/miR expression in a cancer sample to a normal baseline. We also conducted a second evaluation in order to test DEXTER's ability to extract expression data in diseases from the text without the limitations imposed by BioXpress guidelines. Both evaluations are based on comparing DEXTER's output with manually annotated data

sets. The datasets were annotated by domain experts (biologists), who did not participate in the design and implementation of the DEXTER system. The first evaluation used annotations by two researchers who are involved in the BioXpress database design. The second evaluation was based on annotations from a researcher who has considerable experience in biological curation and annotation.

### 6.6.1 Experimental Setup

***BioXpress-based Evaluation:*** For this evaluation, we randomly selected 100 abstracts related to a class of genes (glycosyltransferases) and 100 abstracts related to microRNAs. Since BioXpress is concerned only with expression information in the context of cancer and not other diseases, the abstracts in this evaluation set were selected if they contained some term likely to indicate cancer (e.g., tumor, malignant, cancer, carcinoma, etc.).

To select the microRNA abstracts, we first used the PubMed query "microRNA[TIAB] OR miRNA[TIAB] OR miR[TIAB]", which returned more than 60,000 abstracts. We further filtered and selected only those abstracts that mention a disease as detected by PubTator. Next, we select only those abstracts, which contain certain expression words/phrases such as "expression", "level" etc., which reduces the number of abstracts to 28,067 abstracts. For selecting glycosyltransferase abstracts, instead of using a PubMed query as with microRNAs, we identified the abstracts mentioning any of the glycosyltransferases using the PubTator gene database. Then, as before, we selected a subset of abstracts that contained the expression words/phrases and a disease mention, which yielded 10,278 abstracts. Finally, we randomly selected 200 abstracts from these two sets with an equal number from each set.

Annotators marked the selected abstracts as relevant or not-relevant based on whether they met the criteria for inclusion in the BioXpress database. 90 of the 200 abstracts were annotated as relevant. When an abstract was annotated as relevant, the annotators also identified the associated disease, differentially expressed gene/microRNA and the expression level.

***Second Evaluation:*** The first evaluation only considered cases where gene expression was compared between cancer and normal samples. Therefore, we conducted a second evaluation in order to test more general applications of our text-mining tool. We randomly selected 100 abstracts (divided equally among genes and microRNAs) as an evaluation set following the same procedure for abstract selection used in the first evaluation. This time the set of gene-related abstracts was not limited to glycosyltransferase genes but considered any gene. As before, the annotator marked all the expression information: expressed gene/microRNA, expression level, and associated disease, which resulted in 169 annotated instances. In addition, if the annotator believed it was an explicit comparison of expression level between two different samples, then the annotator also marked the two compared entities.

### 6.6.2 Results

***BioXpress-based Evaluation:*** We ran DEXTER on the first evaluation set and only considered output appropriate for BioXpress i.e. DEXTER output is considered if: (i) the disease is cancer (as determined by Disease Ontology) and (ii) there is an explicit/implicit comparison of expression in cancer samples to normal samples. Therefore, our system identifies Type A sentences, where one of the Compared entities (CEs) has a modifier phrase suggesting it is a control sample (frame-of-

reference flag *Control*) and Type B sentences, where there is an implicit comparison to control (frame-of-reference flag *Control_Implicit*).

An instance was considered to be a True Positive (TP) only when every individual component (expressed gene/microRNA, expression level, and associated cancer) of DEXTER's output matched the corresponding components in the annotation. Thus, an instance can be marked as False Positive (FP) or False Negative (FN) even if just one of the components (e.g., disease) of DEXTER's output did not match the annotation. Table 6.1 (row 1) shows the TP, FN, FP, and Precision (P), Recall (R), F-score (F) measures. The performance on the microRNA- and glycosyltransferase-related abstracts were almost the same.

***Second Evaluation:*** We ran DEXTER on the second evaluation set and compared the output (without any BioXpress specific restrictions) with the annotations. Similar to in the first evaluation, an instance was considered to be True Positive (TP) only when every component of DEXTER's output, including the compared entities, matched the corresponding component in the annotation. Table 6.1 (row 2) shows the TP, FN, FP, and Precision (P), Recall (R), F-score (F) measures for the second evaluation.

| | True Positive | False Positive | False Negative | Precision | Recall | F-score |
|---|---|---|---|---|---|---|
| BioXpress-based Evaluation | 77 | 5 | 15 | 93.90 | 83.69 | 88.51 |
| Second Evaluation | 126 | 13 | 43 | 90.06 | 74.56 | 81.81 |

Table 6.1: DEXTER's Evaluation Results

### 6.6.3 Error Analysis

We conducted an error analysis with the goal of improving our system. We noticed errors were due to mis-parsing, errors in disease detection, the presence of anaphora or lack of patterns. An example of the last type is shown in Example 11. We were unable to capture this case as a Type A relation because in this case, the comparison spans two separate clauses. Because our current set of patterns relies entirely on parsing, it is not possible for the existing system to capture comparisons (hence Type A relations) where the two compared entities appear in different clauses.

> Example 11: Normal human colon cells express low levels of LEF1 and high levels of miR26b; however, human colon cancer cells have decreased miR26b expression and increased LEF1 expression. [PMID: 24785257]

Another type of error involved cases missed due to insufficient triggers. For example, consider the sentence (Example 12), which appears in an abstract used for the second evaluation. The implicit comparison in this sentence was missed by DEXTER because it does not use words like "after" or "following" as Entity Separators. Notice the comparison here is before and after an event, typically a treatment course. This example requires adding new triggers for entity separators, such as "after", and "following". Note that while the precision is roughly the same in both evaluations, the recall is lower in the second evaluation. We believe this might be due

to the stricter guidelines adopted in the first evaluation and more errors encountered due to a greater variety of sentence structures in the second evaluation.

> Example 12: Plasma concentrations of miR-208 increased significantly ($P < 0.0001$) after isoproterenol-induced myocardial injury and showed a similar time course to the concentration of cTnI, a classic biomarker of myocardial injury. [PMID: 19696117]

Another class of false negatives involved sentences where DEXTER were unable to infer the compared aspect/entity from context (elsewhere in the abstract). These include cases where either the CA/CE is not mentioned in the sentence (as in Example 13a) or mentioned as anaphora and requires anaphora/reference resolution (as in Example 13b). For instance, in Example 13a, DEXTER failed to extract the Type A relation, since the compared entities ("PNI tumors", "non-PNI-tumors") was mentioned in a previous sentence. In example 13b, we correctly extracted the Type A relation from the sentence but were unable to extract the microRNAs ("miR-192, miR-194, and miR-215") being referred to in the compared aspect argument ("same microRNA").

> Example 13a: The most differently expressed microRNA was miR-224. [PMID: 18459106]

> Example 13b: The same microRNAs were detected at high levels in normal colon tissue but were severely reduced in many colon cancer samples [PMID: 19074875]

## 6.7 Conclusion

In this chapter, we have described DEXTER, a text-mining tool for the extraction of gene and microRNA expression in disease samples. We have considered two types of sentences indicative of such expression information with (Type A) or

without (Type B) an explicit comparison. From comparative (Type A) sentences we also extract the scenarios in which the expression of the gene/microRNA is contrasted (e.g. disease vs. control). This is particularly useful in capturing the classes of differential expression analyses relevant to the processes of neoplastic transformation and progression such as expression in cancer vs. respective normal tissue, high grade vs. low grade samples, metastasis vs. primary cancer, etc. DEXTER is a direct application of the comparison RE system described in Chapter 5. Additionally, to capture other expression statements (Type B), we have developed a new relation "found_in" based on our relation extraction framework. microRNA expression in disease information extracted by DEXTER in addition to the role of microRNA in diseases extracted by miRiaD [23] (described in Chapter 3) provides a comprehensive resource for microRNA-disease associations.

We have conducted two different evaluations to measure the efficacy of DEXTER. The first evaluation focused on differential gene/microRNA expression in cancer vs. normal samples; the second was more general, covering any description of differential gene/microRNA expression in the context of disease. The system achieved an F-scores of 88.51 and 81.81% for the first and second evaluations, respectively. DEXTER's text-mined results can be used to streamline and accelerate the curation of expression databases such as miR2Disease [20], dbDEMC [92,93]. DEXTER's results have already been integrated into the literature-portion of BioXpress [28], an experimental expression database.

# Chapter 7

# CONCLUSION

Biological entities such as genes, proteins, and microRNAs (miRs) are critical players in various biological/cellular processes and diseases. To understand their impact on cell biology, it is important to know how they are related to their environment. Relationships between these entities and biological concepts (process, disease, pathway, etc.) form a significant part of biomedical knowledge bases. However, a large portion of this information is buried in scientific literature as unstructured text. Development of text-mining (TM) tools to extract such information is important to reduce curation efforts. Most of the existing relation extraction tools in the biomedical domain focus on genes/proteins and specific types of relations such as protein-protein interaction and post-translation modifications. TM tools for miRs' association with diseases and biological processes are very limited.

The development of TM tools to extract the role of miRs in diseases and biological processes is the primary focus of this dissertation. To develop such TM tools, a major part of this dissertation involves the development of a relation extraction (RE) framework by leveraging linguistics generalization for fast development of various RE systems. We have used this RE framework to develop some general relations such as involvement, regulation, and association (called CAIR relations), which we have found can extract a large portion of connections between miR and disease/process. Additionally, to demonstrate the generality and wide applicability of

the RE framework and CAIR relations, we have developed several other TM tools to extract expression level information of miRs and genes in diseases, and the functional impact of protein phosphorylation.

Below I will briefly describe the different parts of the dissertation and the major contributions.

## 7.1 Thesis Summary and Contributions

- We developed a framework for general biomedical relation extraction enabling the fast development of different biomedical text-mining tools. We initially focused on the extraction of relations that connect miRs to associated biological concepts (biological processes). We found that these associations between miRs and concepts fall into a specific type of relations and a small subset of these relations (association, involvement, and regulation) can cover most of the relationships. We call these relations CAIR (Connection through Association, Involvement, and Regulation). We extended the ideas of Extend Dependency Graph (EDG), which leverages various NLP and linguistic generalization principles to develop a general, extendible, and flexible framework for biomedical relation extraction.

  The major contribution of the relation extraction framework is its generality. Most of the existing relation extraction tools in the biomedical domain focus on specific types of relations such as protein-protein interaction, gene target relations, etc. This general relation extraction framework can be used to quickly develop applications to serve different information needs. We use this relation extraction framework to build a RE system to extract CAIR relations from text. The CAIR RE system has been made available as part of the framework. To test the applicability of CAIR relations, we have developed different text-mining tools

based on the CAIR relation extraction system. The EDG framework and the initial notion of CAIR relations have been published in peer-reviewed conferences [3,32].

- We developed an application called miRiaD [23] (microRNAs in association with Disease), based on the CAIR relation extraction system. miRiaD is a text-mining tool that automatically extracts the role of microRNAs in diseases from the literature. Specifically miRiaD extracts various types of miR-disease associations such as (1) miR associations to disease outcome (poor prognosis, survival, etc.), (2) miR association to disease or cellular processes (apoptosis, metastasis, etc.), and (3) miR connections to disease via biomarker and therapeutic target relations. Using miRiaD, we developed a comprehensive microRNA-disease association resource/database and interface that can be used to answer various questions on microRNA's direct and indirect role in diseases. This work was published in the Journal of Biomedical Semantics [23]. The text-mined results of miRiaD has been made available through emiRIT [34], an informatics portal with mined microRNAs in biological networks.

- To demonstrate the wide applicability of our EDG-based relation extraction framework and CAIR relations, we developed a text-mining tool to extract the functional impact of protein phosphorylation. Protein phosphorylation on different sites has functional implications on the substrate (phosphorylated protein) properties. This tool extracts the impact of protein phosphorylation on (1) substrate's interaction with other proteins (binding partners), (2) alternative

subcellular location of the substrate, (3) subsequent further post-translational modification (acetylation, ubiquitination, etc.) of the substrate. This work is a direct application of our EDG-based relation extraction framework and also takes advantage of the CAIR relations, which is used to capture functional impact. Several protein-specific RE systems such as protein-protein interaction, subcellular localization, and post-translation modification relations were developed to capture the various impacted protein properties. This work demonstrates the generality of the EDG framework and CAIR relations by going beyond the extraction of relations involving miRs. A manuscript describing our work on phosphorylation impact is in preparation and we plan to it submit to a Journal soon.

- We developed a comprehensive system to automatically identify comparative structures from the text, which is essential for the development of our TM tool to extract differential expression level of genes and miRs in diseases. Biomedical researchers conduct experiments to validate their hypotheses and infer associations between biological concepts and entities, such as miR, genes, mutation, and disease or therapy and outcome. In such studies, researchers make observations under two different scenarios (e.g., disease sample vs. control sample). When the differences between the groups are statistically significant, the association can be inferred. In addition to be integral for the development of the TM tool to extract differential expression level of genes and miRs in diseases, the comparison extraction system extends the CAIR RE system as **Association** relations, one of our CAIR relations can be inferred from such comparison relations.

We have developed patterns based on our EDG-based relation extraction framework to identify comparison sentences and also extract the various components (compared aspect, compared entities, and scale of the comparison). The developed system identifies explicit comparative structures at the sentence level, where all the components of the comparison are present in the sentence. To the best of our knowledge, ours is the only work that attempts to cover a wide range of comparisons, capture all comparison components, and does not impose any restrictions on the type of compared entities. This work has been published in the ACL 2017 BioNLP Workshop [25].

- While developing miRiaD, we noticed that certain associations between a miR and disease such as expression level in disease sample might not necessarily imply that the miR has a role in the disease but are important to capture. We developed a tool called DEXTER [24], which extracts text evidence of miRs expression level in a diseased state (e.g. expression level of miR in cancer) compared to a normal/non-diseased state. We used our relation extraction framework to extract "found_in" relations from the text, which in addition to the comparisons relation is used in the development of DEXTER. To demonstrate the wide applicability and generality of comparison and "found_in" RE systems, we extend DEXTER to the extraction of differential expression of genes in disease as well. This work was published in the Journal of Biological Databases and Curation [24].

  Additionally, we collaborated with researchers at George Washington University (GWU) involved in the BioXpress [28] database (experimental expression data) and incorporated the automated text-mined output of DEXTER.

Linking such experimental data to existing knowledge from literature will be very useful for researchers. As a result of this collaboration, we processed the entire MEDLINE literature for expression information, and 24, 416 entries were made available for curation into BioXpress. Different works have been published as a result of this integration of DEXTER data with BioXpress such as the Identification of key differentially expressed MicroRNAs in cancer patients through pan-cancer analysis [40] and OncoMX: A Knowledgebase for Exploring Cancer Biomarkers in the Context of Related Cancer and Healthy Data [41].

In addition to the above text-mining (TM) tools developed, several TM tools have developed by others using the EDG-based RE framework and CAIR relations. These include (1) DiMeX [9], a text mining system that finds associations between mutations and diseases from biomedical text, (2) A text mining system called eGARD [33] that extracts the impact of genomic anomalies on drug responses biomedical text, (3) In development tool to extract glycosylation relations of proteins from literature, (4) In development tool to extract the clinical utility of mutation in Alzheimer's disease, and (5) emiRIT [145], an informatics portal with mined microRNAs in biological networks that incorporates text mined results from miRiaD and Dexter along with certain improvement (detection of biological processes) and additions (identification of extracellular miRs) using the EDG-based RE framework.

## 7.2 Future Work

The work in the dissertation can be expanded in a few possible directions.

The EDG-based relation extraction framework currently allows for lexico-syntactic patterns to be written based on part-of-speech (POS) tags, lemma, and

syntactic dependencies. Since most of the text-mining tools need arguments of the relations (CAIR, PPI, etc.) to be of certain types, argument typing is a fundamental part of the development of any downstream TM tools. In the future, we hope to integrate typing information into our framework to allow for the flexibility for patterns to be written on entity types (protein, genes, miR, etc.). The thematic dependency representation motivated by EDG [3] and used in all our RE systems can enable machine learning (ML) applications to generalize more easily as shown in PPI extraction [3,4], chemical disease relation extraction [35]. In the future, we wish to explore the numbered argument representation using our RE framework to develop different ML based RE systems to extract mutation-disease association and glycosylation relations.

All of our TM tools described in the dissertation are applied to Medline abstracts only. The systems could be extended to apply on PMC full-length open access articles, as full-length articles will contain more information than abstracts. This would involve extracting manuscript text, table, figures, and supplementary information from PMC XML dumps. Our various text mining tools could be applied to full-length manuscript text with minor changes.

In this dissertation, we have developed a system for extracting the functional impact of phosphorylation on the substrate's alternative subcellular locations, interaction with binding partners, and further modifications. In this future, we wish to extend the system to cover the phosphorylation impact on other protein properties such as substrate's activation or down-regulation, impact on substrate biological processes and pathways, and protein structure. Additionally, we would like to extract

the functional impact of other post-translational modifications such as acetylation, glycosylation, and ubiquitination.

## REFERENCES


1. Krallinger M, Leitner F, Rodriguez-Penagos C, Valencia A. Overview of the protein-protein interaction annotation extraction task of BioCreative II. Genome Biol. 2008;9 Suppl 2: S4.

2. Tikk D, Thomas P, Palaga P, Hakenberg J, Leser U. A comprehensive benchmark of kernel methods to extract protein-protein interactions from literature. PLoS Comput Biol. 2010;6: e1000837.

3. Peng Y, Gupta S, Wu CH, Vijay-Shanker K. An extended dependency graph for relation extraction in biomedical texts. ACL-IJCNLP 2015. 2015; 21.

4. Peng Y, Lu Z. Deep learning for extracting protein-protein interactions from biomedical literature. arXiv [cs.CL]. 2017. Available: http://arxiv.org/abs/1706.01556

5. Torii M, Li G, Li Z, Oughtred R, Diella F, Celen I, et al. RLIMS-P: an online text-mining tool for literature-based extraction of protein phosphorylation information. Database . 2014;2014. doi:10.1093/database/bau081

6. Dinkel H, Chica C, Via A, Gould CM, Jensen LJ, Gibson TJ, et al. Phospho.ELM: a database of phosphorylation sites--update 2011. Nucleic Acids Res. 2011;39: D261-7.

7. Xu Y, Teng D, Lei Y. MinePhos: a literature mining system for protein phoshphorylation information extraction. IEEE/ACM Trans Comput Biol Bioinform. 2012;9: 311–315.

8. Pyysalo S, Ohta T, Rak R, Sullivan D, Mao C, Wang C, et al. Overview of the ID, EPI and REL tasks of BioNLP Shared Task 2011. BMC Bioinformatics. 2012;13 Suppl 11: S2.

9. Mahmood ASMA, Wu T-J, Mazumder R, Vijay-Shanker K. DiMeX: A Text Mining System for Mutation-Disease Association Extraction. PLoS One. 2016;11: e0152725.

10. Allot A, Peng Y, Wei C-H, Lee K, Phan L, Lu Z. LitVar: a semantic search engine for linking genomic variant data in PubMed and PMC. Nucleic Acids Res. 2018;46: W530–W536.

11. Singhal A, Simmons M, Lu Z. Text mining for precision medicine: automating disease-mutation relationship extraction from biomedical literature. J Am Med Inform Assoc. 2016;23: 766–772.

12. Ravikumar KE, Wagholikar KB, Li D, Kocher J-P, Liu H. Text mining facilitates database curation - extraction of mutation-disease associations from Bio-medical literature. BMC Bioinformatics. 2015;16: 185.

13. Kim J-D, Wang Y. Overview of Genia event task in BioNLP shared task 2011. [cited 13 Oct 2020]. Available: https://www.aclweb.org/anthology/W11-1802.pdf

14. Cejuela JM, Vinchurkar S, Goldberg T, Prabhu Shankar MS, Baghudana A, Bojchevski A, et al. LocText: relation extraction of protein localizations to assist database curation. BMC Bioinformatics. 2018;19: 15.

15. Zheng W, Blake C. Using distant supervised learning to identify protein subcellular localizations from full-text scientific articles. J Biomed Inform. 2015;57: 134–144.

16. Su P, Li G, Wu C, Vijay-Shanker K. Using distant supervision to augment manually annotated data for relation extraction. PLoS One. 2019;14: e0216913.

17. Arighi C, Krallinger M, Leitner F. BioCreative - Track 5: Text mining chemical-protein interactions. [cited 13 Oct 2020]. Available: http://www.biocreative.org/tasks/biocreative-vi/track-5/

18. Peng Y, Rios A, Kavuluru R, Lu Z. Extracting chemical-protein relations with ensembles of SVM and deep learning models. Database (Oxford). 2018;2018. doi:10.1093/database/bay073

19. Antunes R, Matos S. Extraction of chemical-protein interactions from the literature using neural networks and narrow instance representation. Database (Oxford). 2019;2019. doi:10.1093/database/baz095

20. Jiang Q, Wang Y, Hao Y, Juan L, Teng M, Zhang X, et al. miR2Disease: a manually curated database for microRNA deregulation in human disease. Nucleic Acids Res. 2009;37: D98-104.

21. Xie B, Ding Q, Han H, Wu D. miRCancer: a microRNA-cancer association database constructed by text mining on literature. Bioinformatics. 2013;29: 638–644.

22. Li Y, Qiu C, Tu J, Geng B, Yang J, Jiang T, et al. HMDD v2.0: a database for experimentally supported human microRNA and disease associations. Nucleic Acids Res. 2014;42: D1070-4.

23. Gupta S, Ross KE, Tudor CO, Wu CH, Schmidt CJ, Vijay-Shanker K. miRiaD: A Text Mining Tool for Detecting Associations of microRNAs with Diseases. J Biomed Semantics. 2016;7: 9.

24. Gupta S, Dingerdissen H, Ross KE, Hu Y, Wu CH, Mazumder R, et al. DEXTER: Disease-Expression Relation Extraction from Text. Database . 2018;2018. doi:10.1093/database/bay045

25. Gupta S, Mahmood ASMA, Ross K, Wu C, Vijay-Shanker K. Identifying Comparative Structures in Biomedical Text. BioNLP 2017. Stroudsburg, PA, USA: Association for Computational Linguistics; 2017. doi:10.18653/v1/w17-2326

26. Li G, Ross KE, Arighi CN, Peng Y, Wu CH, Vijay-Shanker K. miRTex: A Text Mining System for miRNA-Gene Relation Extraction. PLoS Comput Biol. 2015;11: e1004391.

27. Naeem H, Küffner R, Csaba G, Zimmer R. miRSel: automated extraction of associations between microRNAs and genes from the biomedical literature. BMC Bioinformatics. 2010;11: 135.

28. Wan Q, Dingerdissen H, Fan Y, Gulzar N, Pan Y, Wu T-J, et al. BioXpress: an integrated RNA-seq-derived gene expression database for pan-cancer analysis. Database . 2015;2015. doi:10.1093/database/bav019

29. The Cancer Genome Atlas Research Network, Weinstein JN, Collisson EA, Mills GB, Mills Shaw KR, Ozenberger BA, et al. The Cancer Genome Atlas Pan-Cancer analysis project. Nat Genet. 2013;45: 1113–1120.

30. Zhang J, Baran J, Cros A, Guberman JM, Haider S, Hsu J, et al. International Cancer Genome Consortium Data Portal--a one-stop shop for cancer genomics data. Database . 2011;2011: bar026.

31. Petryszak R, Keays M, Tang YA, Fonseca NA, Barrera E, Burdett T, et al. Expression Atlas update—an integrated database of gene and protein expression

in humans, animals and plants. Nucleic Acids Res. 2015. doi:10.1093/nar/gkv1045

32. Gupta S, Tudor CO, Wu CH, Schmidt CJ, Vijay-Shanker K. Automatically Identifying Biological Functions of microRNAs from the Literature. 6th International Symposium on Semantic Mining in Biomedicine (SMBM 2014). 2014. pp. 75–78.

33. Mahmood ASMA, Rao S, McGarvey P, Wu C, Madhavan S, Vijay-Shanker K. eGARD: Extracting associations between genomic anomalies and drug responses from text. PLoS One. 2017;12: e0189663.

34. Roychowdhury D, Gupta S, Qin X, Arighi CN, Vijay-Shanker K. emiRIT: A text-mining based resource for microRNA information. Cold Spring Harbor Laboratory. 2020. p. 2020.11.05.370593. doi:10.1101/2020.11.05.370593

35. Peng Y, Wei C-H, Lu Z. Improving chemical disease relation extraction with rich features and weakly labeled data. J Cheminform. 2016;8: 53.

36. Schabes Y. Stochastic Lexicalized Tree-adjoining Grammars. Proceedings of the 14th Conference on Computational Linguistics - Volume 2. Stroudsburg, PA, USA: Association for Computational Linguistics; 1992. pp. 425–432.

37. Nativio R, Lan Y, Donahue G, Sidoli S, Berson A, Srinivasan AR, et al. An integrated multi-omics approach identifies epigenetic alterations associated with Alzheimer's disease. Nat Genet. 2020;52: 1024–1035.

38. Lee H-T, Oh S, Ro DH, Yoo H, Kwon Y-W. The key role of DNA methylation and histone acetylation in epigenetics of atherosclerosis. J Lipid Atheroscler. 2020;9: 419.

39. Junqueira SC, Centeno EGZ, Wilkinson KA, Cimarosti H. Post-translational modifications of Parkinson's disease-related proteins: Phosphorylation, SUMOylation and Ubiquitination. Biochim Biophys Acta Mol Basis Dis. 2019;1865: 2001–2007.

40. Hu Y, Dingerdissen H, Gupta S, Kahsay R, Shanker V, Wan Q, et al. Identification of key differentially expressed MicroRNAs in cancer patients through pan-cancer analysis. Comput Biol Med. 2018;103: 183–197.

41. Dingerdissen HM, Bastian F, Vijay-Shanker K, Robinson-Rechavi M, Bell A, Gogate N, et al. OncoMX: A knowledgebase for exploring cancer biomarkers in the context of related cancer and healthy data. JCO Clin Cancer Inform. 2020;4: 210–220.

42. Charniak E. A Maximum-entropy-inspired Parser. Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference. Stroudsburg, PA, USA: Association for Computational Linguistics; 2000. pp. 132–139.

43. Charniak E, Johnson M. Coarse-to-fine N-best Parsing and MaxEnt Discriminative Reranking. Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. Stroudsburg, PA, USA: Association for Computational Linguistics; 2005. pp. 173–180.

44. Mcclosky D. Any Domain Parsing: Automatic Domain Adaptation for Natural Language Parsing. Brown University. 2010. Available: http://dl.acm.org/citation.cfm?id=2020153

45. Manning CD, Surdeanu M, Bauer J, Finkel JR, Bethard S, McClosky D. The stanford corenlp natural language processing toolkit. ACL (System Demonstrations). 2014. pp. 55–60.

46. De Marneffe M-C, Dozat T, Silveira N, Haverinen K, Ginter F, Nivre J, et al. Universal Stanford dependencies: A cross-linguistic typology. LREC. 2014. pp. 4585–4592.

47. ˇZeljko Agic MJA, Atutxa A, Bosco C, Choi J, de Marneffe M-C, Dozat T, et al. Universal dependencies 1.1. LINDAT/CLARIN digital library at Institute of Formal and Applied Linguistics, Charles University in Prague. 2015. Available: http://universaldependencies.org/

48. Palmer M, Gildea D, Kingsbury P. The Proposition Bank: An Annotated Corpus of Semantic Roles. Comput Linguist. 2005;31: 71–106.

49. Bonial C. English PropBank Annotation Guidelines. [cited 14 Oct 2020]. Available: http://verbs.colorado.edu/propbank/EPB-Annotation-Guidelines.pdf

50. SemgrexPattern (Stanford JavaNLP API). [cited 16 Jan 2017]. Available: http://nlp.stanford.edu/nlp/javadoc/javanlp/edu/stanford/nlp/semgraph/semgrex/SemgrexPattern.html

51. Koopman HJ, Sportiche D, Stabler E. An introduction to syntactic analysis and theory. London, England: Blackwell; 2014.

52. Hu ZZ, Narayanaswamy M, Ravikumar KE, Vijay-Shanker K, Wu CH. Literature mining and database annotation of protein phosphorylation using a rule-based system. Bioinformatics. 2005;21: 2759–2765.

53. Narayanaswamy M, Ravikumar KE, Vijay-Shanker K. Beyond the clause: extraction of phosphorylation information from medline abstracts. Bioinformatics. 2005;21 Suppl 1: i319-27.

54. Peng Y, Torii M, Wu CH, Vijay-Shanker K. A generalizable NLP framework for fast development of pattern-based biomedical relation extraction systems. BMC Bioinformatics. 2014;15: 285.

55. Baker CF, Fillmore CJ, Lowe JB. The Berkeley FrameNet Project. Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1. Stroudsburg, PA, USA: Association for Computational Linguistics; 1998. pp. 86–90.

56. Arighi CN, Lu Z, Krallinger M, Cohen KB, Wilbur WJ, Valencia A, et al. Overview of the BioCreative III Workshop. BMC Bioinformatics. 2011;12 Suppl 8: S1.

57. Kim J-D, Wang Y, Yasunori Y. The Genia event extraction shared task, 2013 edition - overview. [cited 14 Oct 2020]. Available: https://www.aclweb.org/anthology/W13-2002.pdf

58. Segura-Bedmar I, Martínez P, Herrero-Zazo M. SemEval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (DDIExtraction 2013). Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013). Atlanta, Georgia, USA: Association for Computational Linguistics; 2013. pp. 341–350.

59. Chen ES, Hripcsak G, Xu H, Markatou M, Friedman C. Automated acquisition of disease drug knowledge from biomedical and clinical documents: an initial study. J Am Med Inform Assoc. 2008;15: 87–98.

60. Doughty E, Kertesz-Farkas A, Bodenreider O, Thompson G, Adadey A, Peterson T, et al. Toward an automatic method for extracting cancer- and other disease-related point mutations from the biomedical literature. Bioinformatics. 2011;27: 408–415.

61. Vlachos A, Craven M. Biomedical event extraction from abstracts and full papers using search-based structured prediction. BMC Bioinformatics. 2012;13 Suppl 11: S5.

62. Riedel S, McClosky D, Surdeanu M, McCallum A, Manning CD. Model Combination for Event Extraction in BioNLP 2011. Proceedings of the BioNLP

Shared Task 2011 Workshop. Stroudsburg, PA, USA: Association for Computational Linguistics; 2011. pp. 51–55.

63. Björne J, Salakoski T. Generalizing Biomedical Event Extraction. Proceedings of the BioNLP Shared Task 2011 Workshop. Stroudsburg, PA, USA: Association for Computational Linguistics; 2011. pp. 183–191.

64. Airola A, Pyysalo S, Björne J, Pahikkala T, Ginter F, Salakoski T. All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning. BMC Bioinformatics. 2008;9 Suppl 11: S2.

65. Miwa M, Sætre R, Miyao Y, Tsujii J. Protein–protein interaction extraction by leveraging multiple kernels and parsers. Int J Med Inform. 2009;78: e39–e46.

66. Bui Q-C, Katrenko S, Sloot PMA. A hybrid approach to extract protein–protein interactions. Bioinformatics. 2011;27: 259–265.

67. Kim S, Yoon J, Yang J, Park S. Walk-weighted subsequence kernels for protein-protein interaction extraction. BMC Bioinformatics. 2010;11: 107.

68. Pyysalo S, Airola A, Heimonen J, Björne J, Ginter F, Salakoski T. Comparative analysis of five protein-protein interaction corpora. BMC Bioinformatics. 2008;9 Suppl 3: S6.

69. Peng Y, Arighi C, Wu CH, Vijay-Shanker K. Extended dependency graph for BioC-compatible protein-protein interaction (PPI) passage detection in full-text articles. Available: http://www.biocreative.org/media/store/files/2015/BCV2015_paper_10.pdf

70. Xu J, Wu Y, Zhang Y, Wang J, Liu R, Wei Q, et al. UTH-CCB@ BioCreative V CDR task: identifying chemical-induced disease relations in biomedical text. Proceedings of the Fifth BioCreative Challenge Evaluation Workshop. biocreative.org; 2015. pp. 254–259.

71. Gu JH, Qian LH, Zhou GD. Chemical-induced disease relation extraction with lexical features. Proceedings of the fifth BioCreative Challenge Evaluation Workshop BioCreative Organizing Committee Sevilla, Spain. biocreative.org; 2015. pp. 220–225.

72. Jiang Z, Jin L, Li L, Qin M, Qu C, Zheng J, et al. A CRD-WEL system for chemical-disease relations extraction. The fifth BioCreative challenge evaluation workshop. pdfs.semanticscholar.org; 2015. pp. 317–326.

73. Cohen KB, Verspoor K, Johnson HL, Roeder C, Ogren PV, Baumgartner WA Jr, et al. HIGH-PRECISION BIOLOGICAL EVENT EXTRACTION: EFFECTS OF SYSTEM AND OF DATA. Comput Intell. 2011;27: 681–701.

74. Hakenberg J, Leaman R, Vo NH, Jonnalagadda S, Sullivan R, Miller C, et al. Efficient extraction of protein-protein interactions from full-text articles. IEEE/ACM Trans Comput Biol Bioinform. 2010;7: 481–494.

75. Kilicoglu H, Bergler S. Adapting a General Semantic Interpretation Approach to Biological Event Extraction. Proceedings of the BioNLP Shared Task 2011 Workshop. Stroudsburg, PA, USA: Association for Computational Linguistics; 2011. pp. 173–182.

76. Kim J-J, Rebholz-Schuhmann D. Improving the extraction of complex regulatory events from scientific text by using ontology-based inference. J Biomed Semantics. 2011;2 Suppl 5: S3.

77. Peng Y, Tudor CO, Torii M, Wu CH, Vijay-Shanker K. iSimp: A sentence simplification system for biomedicail text. 2012 IEEE International Conference on Bioinformatics and Biomedicine. 2012. pp. 1–6.

78. Tudor CO, Vijay-Shanker K. RankPref: Ranking Sentences Describing Relations between Biomedical Entities with an Application. Available: http://www.aclweb.org/anthology/W12-2420

79. Chen J, Shanker VK. Automated Extraction of Tags from the Penn Treebank. In: Bunt H, Carroll J, Satta G, editors. New Developments in Parsing Technology. Springer Netherlands; 2004. pp. 73–89.

80. Karin Kipper Schuler U of P, Authors. VerbNet: A broad-coverage, comprehensive verb lexicon. University of Pennsylvania. 2005. Available: http://repository.upenn.edu/dissertations/AAI3179808/

81. Kipper K, Korhonen A, Ryant N, Palmer M. Extending VerbNet with novel verb classes. Proceedings of LREC. pdfs.semanticscholar.org; 2006. p. 1.

82. Lippincott T, Rimell L, Verspoor K, Korhonen A. Approaches to verb subcategorization for biomedicine. J Biomed Inform. 2013;46: 212–227.

83. Rimell L, Lippincott T, Verspoor K, Johnson HL, Korhonen A. Acquisition and evaluation of verb subcategorization resources for biomedicine. J Biomed Inform. 2013;46: 228–237.

84. Blenkiron C, Miska EA. miRNAs in cancer: approaches, aetiology, diagnostics and therapy. Hum Mol Genet. 2007;16 Spec No 1: R106-13.

85. Xu X, Li S, Lin Y, Chen H, Hu Z, Mao Y, et al. MicroRNA-124-3p inhibits cell migration and invasion in bladder cancer cells by targeting ROCK1. J Transl Med. 2013;11: 276.

86. Colangelo T, Fucci A, Votino C, Sabatino L, Pancione M, Laudanna C, et al. MicroRNA-130b promotes tumor development and is associated with poor prognosis in colorectal cancer. Neoplasia. 2013;15: 1218–1231.

87. Jiang B-Y, Zhang X-C, Su J, Meng W, Yang X-N, Yang J-J, et al. BCL11A overexpression predicts survival and relapse in non-small cell lung cancer and is modulated by microRNA-30a and gene amplification. Mol Cancer. 2013;12: 61.

88. Yu X, Zhang W, Ning Q, Luo X. MicroRNA-34a inhibits human brain glioma cell growth by down-regulation of Notch1. J Huazhong Univ Sci Technolog Med Sci. 2012;32: 370–374.

89. Xu Y, Zhao F, Wang Z, Song Y, Luo Y, Zhang X, et al. MicroRNA-335 acts as a metastasis suppressor in gastric cancer by targeting Bcl-w and specificity protein 1. Oncogene. 2012;31: 1398–1407.

90. Wang Q, Huang Z, Ni S, Xiao X, Xu Q, Wang L, et al. Plasma miR-601 and miR-760 are novel biomarkers for the early detection of colorectal cancer. PLoS One. 2012;7: e44398.

91. Li T, Li R-S, Li Y-H, Zhong S, Chen Y-Y, Zhang C-M, et al. miR-21 as an independent biochemical recurrence predictor and potential therapeutic target for prostate cancer. J Urol. 2012;187: 1466–1472.

92. Yang Z, Ren F, Liu C, He S, Sun G, Gao Q, et al. dbDEMC: a database of differentially expressed miRNAs in human cancers. BMC Genomics. 2010;11 Suppl 4: S5.

93. Yang Z, Wu L, Wang A, Tang W, Zhao Y, Zhao H, et al. dbDEMC 2.0: updated database of differentially expressed miRNAs in human cancers. Nucleic Acids Res. 2017;45: D812–D818.

94. Dweep H, Sticht C, Pandey P, Gretz N. miRWalk – Database: Prediction of possible miRNA binding sites by “walking” the genes of three genomes. J Biomed Inform. 2011;44: 839–847.

95. Sethupathy P, Corda B, Hatzigeorgiou AG. TarBase: A comprehensive database of experimentally supported animal microRNA targets. RNA. 2006;12: 192–197.

96. Hsu S-D, Lin F-M, Wu W-Y, Liang C, Huang W-C, Chan W-L, et al. miRTarBase: a database curates experimentally validated microRNA-target interactions. Nucleic Acids Res. 2011;39: D163-9.

97. Griffiths-Jones S, Grocock RJ, van Dongen S, Bateman A, Enright AJ. miRBase: microRNA sequences, targets and gene nomenclature. Nucleic Acids Res. 2006;34: D140-4.

98. Wei C-H, Kao H-Y, Lu Z. PubTator: a web-based text mining tool for assisting biocuration. Nucleic Acids Res. 2013;41: W518-22.

99. Leaman R, Islamaj Dogan R, Lu Z. DNorm: disease name normalization with pairwise learning to rank. Bioinformatics. 2013;29: 2909–2917.

100. Natale DA, Arighi CN, Barker WC, Blake JA, Bult CJ, Caudy M, et al. The Protein Ontology: a structured representation of protein forms and complexes. Nucleic Acids Res. 2011;39: D539-45.

101. Tudor CO, Arighi CN, Wang Q, Wu CH, Vijay-Shanker K. The eFIP system for text mining of protein interaction networks of phosphorylated proteins. Database (Oxford). 2012;2012: bas044.

102. Tudor CO, Ross KE, Li G, Vijay-Shanker K, Wu CH, Arighi CN. Construction of phosphorylation interaction networks by text mining of full-length articles using the eFIP system. Database (Oxford). 2015;2015. doi:10.1093/database/bav020

103. Torii M, Arighi CN, Li G, Wang Q, Wu CH, Vijay-Shanker K. RLIMS-P 2.0: A generalizable rule-based information extraction system for literature mining of protein phosphorylation information. IEEE/ACM Trans Comput Biol Bioinform. 2015;12: 17–29.

104. Zhao Z, Yang Z, Lin H, Wang J, Gao S. A protein-protein interaction extraction approach based on deep neural network. Int J Data Min Bioinform. 2016;15: 145.

105. Li G. Biomedical relation extraction with reduced manual effort. University of Delaware. 2018. Available: http://udspace.udel.edu/handle/19716/23793

106. Ohta T, Pyysalo S, Miwa M, Kim J-D, Tsujii J. Event Extraction for Post-Translational Modifications. [cited 14 Oct 2020]. Available: https://www.aclweb.org/anthology/W10-1903.pdf

107. Ohta T, Pyysalo S, Tsujii J. Overview of the epigenetics and post-translational modifications (EPI) task of BioNLP shared task 2011. Proceedings of BioNLP Shared Task 2011 Workshop. Portland, Oregon, USA: Association for Computational Linguistics; 2011. pp. 16–25.

108. Raghavan P, Fosler-Lussier E, Lai A. Temporal classification of medical events. BioNLP: Proceedings of the 2012 Workshop on Biomedical Natural Language Processing. Montréal, Canada: Association for Computational Linguistics; 2012. pp. 29–37.

109. Miwa M, Sætre R, Kim J-D, Tsujii J. Event extraction with complex event classification using rich features. J Bioinform Comput Biol. 2010;08: 131–146.

110. van der Horn P, Bakker B, Geleijnse G, Korst J, Kurkin S. Determining causal and non-causal relationships in biomedical text by classifying verbs using a naive Bayesian classifier. Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing - BioNLP '08. Morristown, NJ, USA: Association for Computational Linguistics; 2008. doi:10.3115/1572306.1572335

111. Mihăilă C, Ohta T, Pyysalo S, Ananiadou S. BioCause: Annotating and analysing causality in the biomedical domain. BMC Bioinformatics. 2013;14: 2.

112. Subcellular locations. [cited 14 Oct 2020]. Available: https://www.uniprot.org/locations/

113. Orchard S, Ammari M, Aranda B, Breuza L, Briganti L, Broackes-Carter F, et al. The MIntAct project--IntAct as a common curation platform for 11 molecular interaction databases. Nucleic Acids Res. 2014;42: D358-63.

114. Park DH, Blake C. Identifying Comparative Claim Sentences in Full-text Scientific Articles. Proceedings of the Workshop on Detecting Structure in Scholarly Discourse. Stroudsburg, PA, USA: Association for Computational Linguistics; 2012. pp. 1–9.

115. Bresnan JW. Syntax of the Comparative Clause Construction in English. Linguist Inq. 1973;4: 275–343.

116. Friedman C. A General Computational Treatment of the Comparative. Proceedings of the 27th Annual Meeting on Association for Computational

Linguistics. Stroudsburg, PA, USA: Association for Computational Linguistics; 1989. pp. 161–168.

117. Staab S, Hahn U. Comparatives in Context. Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Conference on Innovative Applications of Artificial Intelligence. Providence, Rhode Island: AAAI Press; 1997. pp. 616–621.

118. Jindal N, Liu B. Identifying Comparative Sentences in Text Documents. Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, NY, USA: ACM; 2006. pp. 244–251.

119. Jindal N, Liu B. Mining comparative sentences and relations. AAAI. 2006. Available: https://vvvvw.aaai.org/Papers/AAAI/2006/AAAI06-209.pdf

120. Xu K, Liao SS, Li J, Song Y. Mining Comparative Opinions from Customer Reviews for Competitive Intelligence. Decis Support Syst. 2011;50: 743–754.

121. Ganapathibhotla M, Liu B. Mining Opinions in Comparative Sentences. Proceedings of the 22Nd International Conference on Computational Linguistics - Volume 1. Stroudsburg, PA, USA: Association for Computational Linguistics; 2008. pp. 241–248.

122. Fiszman M, Demner-Fushman D, Lang FM, Goetz P, Rindflesch TC. Interpreting Comparative Constructions in Biomedical Text. Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing. Stroudsburg, PA, USA: Association for Computational Linguistics; 2007. pp. 137–144.

123. Rindflesch TC, Fiszman M. The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text. J Biomed Inform. 2003;36: 462–477.

124. Rindflesch TC, Fiszman M, Libbus B. Semantic Interpretation for the Biomedical Research Literature. In: Chen H, Fuller SS, Friedman C, Hersh W, editors. Medical Informatics. Springer US; 2005. pp. 399–422.

125. The Stanford NLP Group. [cited 13 Oct 2020]. Available: https://nlp.stanford.edu/software/stanford-dependencies.html

126. Greco S, Gorospe M, Martelli F. Noncoding RNA in age-related cardiovascular diseases. J Mol Cell Cardiol. 2015;83: 142–155.

127. Moura J, Børsheim E, Carvalho E. The Role of MicroRNAs in Diabetic Complications-Special Emphasis on Wound Healing. Genes . 2014;5: 926–956.

128. Maciotta S, Meregalli M, Torrente Y. The involvement of microRNAs in neurodegenerative diseases. Front Cell Neurosci. 2013;7: 265.

129. Gori M, Arciello M, Balsano C. MicroRNAs in nonalcoholic fatty liver disease: novel biomarkers and prognostic tools during the transition from steatosis to hepatocarcinoma. Biomed Res Int. 2014;2014: 741465.

130. Chapman CG, Pekow J. The emerging role of miRNAs in inflammatory bowel disease: a review. Therap Adv Gastroenterol. 2015;8: 4–22.

131. Nalejska E, Mączyńska E, Lewandowska MA. Prognostic and predictive biomarkers: tools in personalized oncology. Mol Diagn Ther. 2014;18: 273–284.

132. Barrett T, Edgar R. Mining microarray data at NCBI's Gene Expression Omnibus (GEO)*. Methods Mol Biol. 2006;338: 175–190.

133. Parkinson H, Sarkans U, Shojatalab M, Abeygunawardena N, Contrino S, Coulson R, et al. ArrayExpress--a public repository for microarray gene expression data at the EBI. Nucleic Acids Res. 2005;33: D553-5.

134. Liu X, Yu X, Zack DJ, Zhu H, Qian J. TiGER: a database for tissue-specific gene expression and regulation. BMC Bioinformatics. 2008;9: 271.

135. Li H, He Y, Ding G, Wang C, Xie L, Li Y. dbDEPC: a database of differentially expressed proteins in human cancers. Nucleic Acids Res. 2010;38: D658-64.

136. He Y, Zhang M, Ju Y, Yu Z, Lv D, Sun H, et al. dbDEPC 2.0: updated database of differentially expressed proteins in human cancers. Nucleic Acids Res. 2012;40: D964-71.

137. Bauer-Mehren A, Rautschka M, Sanz F, Furlong LI. DisGeNET: a Cytoscape plugin to visualize, integrate, search and analyze gene–disease networks. Bioinformatics. 2010;26: 2924–2926.

138. Piñero J, Queralt-Rosinach N, Bravo À, Deu-Pons J, Bauer-Mehren A, Baron M, et al. DisGeNET: a discovery platform for the dynamical exploration of human diseases and their genes. Database . 2015;2015: bav028.

139. Wang D, Gu J, Wang T, Ding Z. OncomiRDB: a database for the experimentally verified oncogenic and tumor-suppressive microRNAs. Bioinformatics. 2014;30: 2237–2238.

140. Dingerdissen HM, Torcivia-Rodriguez J, Hu Y, Chang T-C, Mazumder R, Kahsay R. BioMuta and BioXpress: mutation and expression knowledgebases for cancer biomarker discovery. Nucleic Acids Res. 2017 [cited 22 Nov 2017]. doi:10.1093/nar/gkx907

141. Zhang J, Baran J, Cros A, Guberman JM, Haider S, Hsu J, et al. International Cancer Genome Consortium Data Portal—a one-stop shop for cancer genomics data. Database . 2011;2011. doi:10.1093/database/bar026

142. Maglott D, Ostell J, Pruitt KD, Tatusova T. Entrez Gene: gene-centered information at NCBI. Nucleic Acids Res. 2011;39: D52-7.

143. Schriml LM, Arze C, Nadendla S, Chang Y-WW, Mazaitis M, Felix V, et al. Disease Ontology: a backbone for disease semantic integration. Nucleic Acids Res. 2012;40: D940-6.

144. Davis AP, Wiegers TC, Rosenstein MC, Mattingly CJ. MEDIC: a practical disease vocabulary used at the Comparative Toxicogenomics Database. Database . 2012;2012: bar065.

145. Mirna_Entities_Relations. [cited 13 Oct 2020]. Available: https://research.bioinformatics.udel.edu/iGRINminer/

## Appendix A

## CONVENTION FOR WRITING RULES

The rule for adding new edges (numbered or otherwise) consists of set of conditions and associated actions. If all the conditions are satisfied (on the Standard Dependency Graph), then a set of nodes are populated. Based on the populated nodes, user can write actions specifying between which nodes he/she wants to add edges. The set of conditions are based on Stanford Semgrex pattern. A semgrex is regular expression matching tool where users can specify a regular expression on Standard Dependencies. These regular expressions can be specified on token node information (lemma, POS tags, word) and dependency edges between token nodes. The sets of conditions (semregex patterns) are applied to the dependency and it populates a set of nodes based on the pattern. Then the actions are applied to add new edges between the named nodes. More details about semregex patterns can be found at the link [5].

Below we will explain our rule writing convention through one simple example below. In this example there are three conditions and two actions. In Cond_1, we specify a Semgrex condition on a token with POS tage VBN and lemma "involve" or "implicate". If this condition is specified that token is assigned to named node N0. Cond_2 specify a dependency condition, where the node N0 is the governor of a relation "nsubjpass" or "nsubj_null" with N1 being the dependent of the relation. If

---

[5] http://nlp.stanford.edu/nlp/javadoc/javanlp/edu/stanford/nlp/semgraph/semgrex/SemgrexPattern.html

such a dependency edge is found, then the dependent token is named N1. Cond_3 specifies another dependency condition, where the node N0 is the governor of a relation “dobj” with N2 being the dependent of the relation. If such a dependency edge is found, then the dependent token is named N2. Action_1 specifies an action of adding a new edge called “arg0” from N0 to N1 and “arg1” from N0 to N2.

Figure B.1 shows the Standard Dependency Graph (SDG) for an example on which this rule is applied and the resulting Extended Dependency Graph (EDG) with added numbered-arguments in Figure B.2. In this case based on conditions the populated named nodes are N0: “involved”, N1: “miR-21”, N2: “proliferation”

```
Sample Rule:
####X is invovlved in Y######
RuleID : involvement_in_1
Cond_1 : {pos:/VBN/;lemma:/(involve|implicate)/}=N0
Cond_2 : {}=N0 >/(nsubjpass|nsubj_null)/ {}=N1
Cond_3 : {}=N0 >/nmod:in/ {}=N2
Action_1 : N0 >> arg0 >> N1
Action_2 : N0 >> arg1 >> N2
```

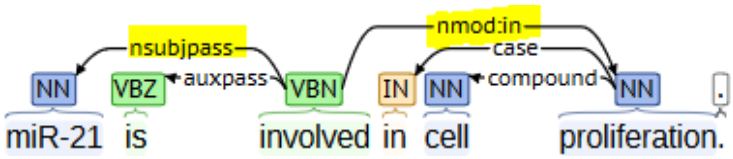


Figure A.1 : Sample SDG before rule application

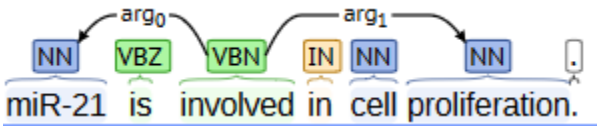


Figure A.2 Sample EDG after application of Rule 1

## Appendix B

## PARSING ERROR CORRECTION: PHASE 0 RULES

In this phase, we try to correct some of errors of either the BLLIP parser or the conversion tool such that the rules in other phases do not have handle such errors. Because these errors are due to incorrect parsing, these cases will be handled before the phase in which we add numbered argument edges. We noticed there were some systematic errors being made.

One such systematic error was the incorrect assignment of *dep* relation edge (instead of *appos*) in sentences which enumerate genes/microRNAs. According to Universal Dependencies [24] "A dependency can be labeled as *dep* when it is impossible to determine a more precise relation. This may be because of a weird grammatical construction, or a limitation in conversion or parsing software". Consider the sentence in Figure B.1 below.

Notice the edges from "microRNAs" to "miR-373" and "miR-520c" are labelled as *dep* instead of *appos* (indicating appositives). This issue is very common and needed to be addressed. We use the information that governor of the *dep* edges has a part-of-speech (POS) of NNS and the dependents (miR-373 and miR-520c) are conjuncts (detected through *conj:and*) to add the edge *appos_added* (highlighted) from "microRNAs" to "miR-373" and "miR-520c" thereby correcting the error. In order to differentiate between *appos* edges added by us and those detected by Stanford tool, we add the edge "appos_added" instead of "appos".

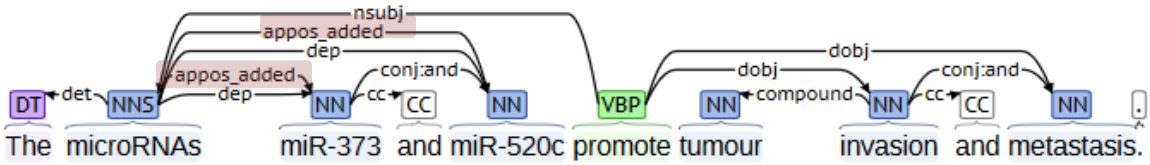


Figure B.1: SDG Error Correction Example

Another systematic parsing error, which we have noticed is the detection of conjuncts in complex sentences. We will explore the correction of such conjuct errors in the future.

## Appendix C

## LIST OF TRIGGERS

Below we provide the different lexical triggers for predicates used in different RE systems (CAIR relations, PPI, etc.) and typing triggers (expression, disease sample etc.) used in different TM tools (miRiaD, DEXTER). Although trigger words are provided in a certain tense in the examples below, the reader should assume all of their textual variations if it exists (tense and nominalized forms). .*X indicates, triggers with suffix X.

### C.1 Triggers for CAIR relations

1. Involvement Triggers

"contribute to", "involved in", "implicated in", "participates in", "plays/has a role", "required for/in", "needed for/in", "important for/in", "effect on", "influence", "affect", "necessary for", "needed for", critical for", "essential for".

2. Regulation Triggers:

"abolish", "accelerate", "activate", "alter", "attenuate", "block", "cause", "change", "control", "decrease", "deregulate", diminish", "disrupts", "down/up-regulate", "elevate", "eliminate", "enhance", "impact", "increase", "induce", "impair" "inhibit", "limit", "maintain", mediate", "modulate", "promote", "reduce", "regulate", "repress", "restrict", "sensitize", "stimulate", "suppress", "suppress", "target"

3. Association Triggers

"associated with", "correlated with/to", "linked to/with", "depended on", "linked to/with"

## C.2 Triggers for Found-in Relation

Found_In Triggers:

Set 1: "found in", "noted in", "detected in", "observed in", "discovered in", "occurred in"

Set 2: "increased in", "decreased in", "over/under-expressed in", "expressed in", "silenced in", "reduced in", "elevated in", "changed in", "regulated in", "up/down-regulated in"

## C.3 Triggers for is-a Relations

Is-a Triggers:

In addition to copular sentences indicated through "is" and "are" and appositives, the following triggers are used to detect Is-a relations:

"serves as", "functions as", "acts as", "knowns as", "known to be"

## C.4 Triggers for Protein-Protein Interaction Relations

bind, interact, associate, dimerize, co-immunoprecipitate, coimmunoprecipitate, hetrodimerize, recruit, affinity, dissociation, complex, .*dimer|.*trimer|.*tramer|.*tamer|.*hexamer|.*nonamer|.*octomer

## C.5 Triggers for Subcellular Localization Relations

localize, translocate, traffic, transport, import, export, relocalize, accumulate, trafficking,

## C.6 Triggers for Post-Translational Modification Relations

phosphorylate, ubiquitinate, acetylate, glycosylate, neddylate, sumoylate

## C.7 Expression Phrase Typing Triggers

“over-expression”, “under-expression”, “expression”, “up-regulation”, “down-regulation”, “overexpression”, “underexpression”, “upregulation”, “downregulation”, “level”, “knockdown”, “elevation”, “production”, “silencing”, “loss”, “gain”, “depletion”, “absence”, “abundance”, “concentration”

## C.8 Disease Sample Phrase Typing Triggers

“tissue”, cell”, “patient”, “sample”, “tumor”, “cancer”, “carcinoma”, “cell line”, “cell-line”, “group”, “blood”, “sera”, “serum”, fluid”, “subset”, “case”, “men”, “women”

NOTE: All valid plural forms are also considered

## C.9 Control Sample Phrase Typing Triggers

control, normal, health, healthy, NC, adjacent, peri-tumoral, peritumoral, non( |- )?(tumor|tumoral|cancerous)

NOTE: All valid plural forms are also considered

### C.10 Expression Level Triggers

1. High Expression Level triggers:

"gain", "increased", "high", "overexpressed", "over-expressed", "positive", "strong", elevated", "significant", "upregulated", "up-regulated"

2. Low Expression Level Triggers:

"loss", "decreased", "low", "underexpressed", "under-expressed", "down-regulated" "downregulated", reduced", "knockdown", "suppressed", "negative", "weak"

## Appendix D

## LEXICO-SYNTACTIC RULES

Below, we provide links to set for lexico-syntactic for the different relations discussed in this dissertation. Each link to a GitHub repository file contains a list of lexico-syntactic rules in the common rule format discussed in Appendix A. Essentially these lexico-syntatic rules add edges (numbered arg0/arg1 or otherwise) consists of conditions and associated actions.

If all the conditions are satisfied (on the Dependency Graph), then a set of nodes are populated. Based on the populated nodes, user can write actions specifying between which nodes he/she wants to add edges. The set of conditions are based on Stanford Semgrex pattern. The sets of conditions (semregex patterns) are applied to the dependency and it populates a set of nodes based on the pattern. Then the actions are applied to add new edges between the named nodes. Each Cond_# is Semregex Pattern Each Action_# adds edges. All files can be found in the GitHub repository:

https://github.com/samirgupta/lexico-syntactic_rules

Below, for each relation we provide the link to the corresponding a GitHub repository file containing all lexico-syntactic rules in the common rule format discussed in Appendix A.

- **Rules for Extra-syntactic relations**
  Link: Extra_syntactic

- **Rules for handling Null-Argument structures**
  Link: Null_Argument

- **Rules for OpenIE**
  Link: OpenIE

- **Rules for CAIR Relation : Involvement**
  Link: CAIR_Involvement

- **Rules for CAIR Relation : Regulation**
  Link: CAIR_Regulation

- **Rules for CAIR Relation : Association**
  Link: CAIR_Association

- **Rules for Protein-Protein Interaction relation**
  Link: Protein-Protein_Interaction

- **Rules for Subcellular Localization relation**
  Link: Subcellular_Localization

- **Rules for Other Post-Translational Modification relation**
  Link: Post-Translational_Modification

- **Rules for Explicit Temporal Impact relation**
  Link: Explicit_Temporal_Impact

- **Rules for Comparison relations**
  Link: Comparison

- **Rules for found_in relation.**
  Link: Found_in

## Appendix E

## ILLUSTRATING THE WORKINGS OF THE RELATION EXTRACTION FRAMEWORK

In this Appendix, we describe the working of the Relation Extraction Framework described in Chapter 2, Section 2.3 through a running example. We will show the output of the different steps in our framework on the example sentence (1) below.

Example 1: MicroRNA-122, a tumor suppressor microRNA that regulates intrahepatic metastasis of hepatocellular carcinoma.

Recall the architecture of the framework depicted in Figure 2.11 in Chapter 2. Given an input text, typically a Medline abstract, we first tokenize and split the text into sentences using the Stanford CoreNLP toolkit. The "Parser" step in the framework takes as input a single sentence and outputs a constituency parse tree using the BLLIP parser. The constituent parse tree for the example sentence (1) is shown in Figure E.1. The second step "Dependency conversion" uses the Stanford conversion tool to convert the above parse tree into syntactic dependencies. The syntactic dependencies obtained after applying the conversion tool on the parse tree is shown in Figure E.2.

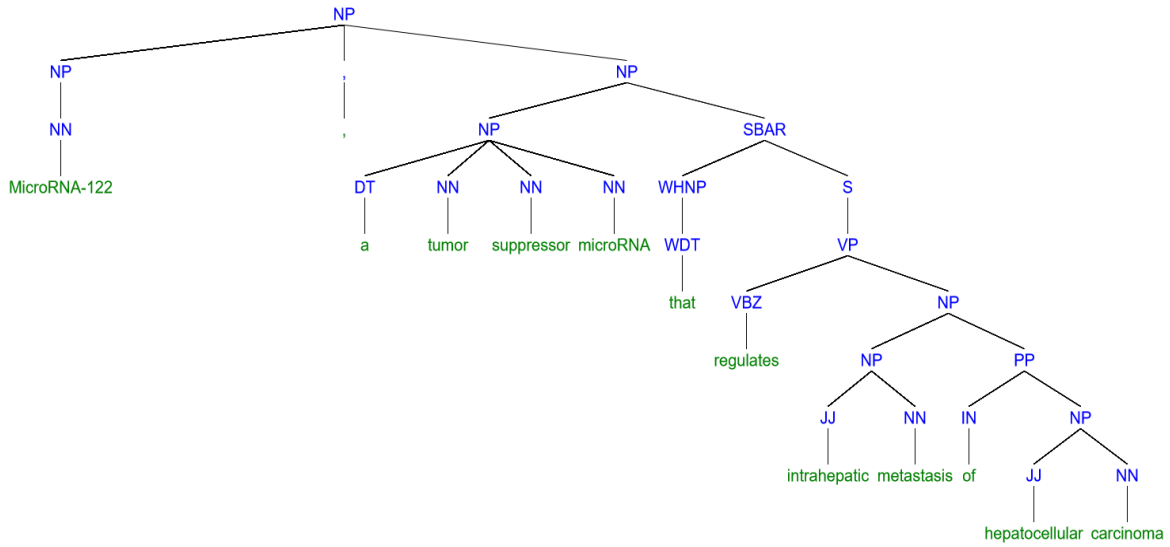


Figure E.1: Example Parse Tree

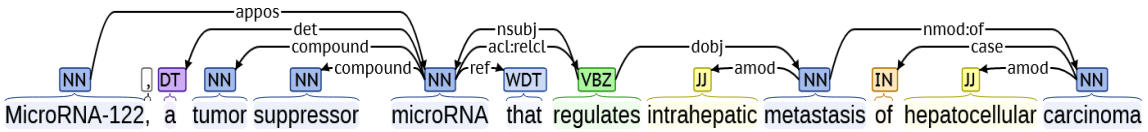


Figure E.2 Example Syntactic Dependencies

Based on this syntactic dependency representation, Semgrex patterns defined in Phases 0 – 3 are applied to add additional edges (is-a, numbered arguments, etc.). In our current example, the "extra-syntactic" Phase 1 rule to add is-a relation we will be applied to capture the Is-a relation between "MicroRNA-122" and "tumor suppressor microRNA". The lexico-syntactic "is-a" Semgrex rule (shown in the textbox below) below will be applied and matched against the syntactic dependency structure using the "Pattern matching step" and the new syntactic dependency with the extra-syntactic link "is-a" will be generated as shown in Figure E.3. The details of the Semgrex rule

format and pattern matching have already been described in Appendix A. In this particular example, the is-a edge is added by following the appositive, *appos* edge.

```
####Appos Rule####
RuleID : isa_2
Cond_0 : {}=N0 >/(appos|appos_added)/ {}=N1
Action_1 : N0 >> is_a >> N1
```

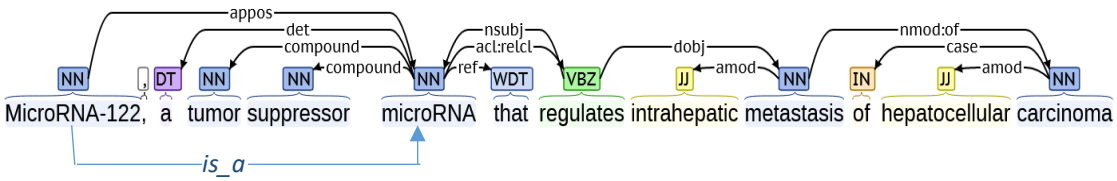


Figure E.3: Dependency Graph after applying Phase 1 set of rules: Is-a

The next set of rules that are applied are Phase 1, RE-specific rules to add numbered-argument edges for user-defined relations. In this particular example, we show the application of adding numbered-argument edges for the CAIR relation of Regulation. The active case of the regulation relation Semgrex rule is shown in the text-box below, which when applied to the Syntactic Dependency in Figure E.3 generates the Extended Dependency Graph with the *arg0* and *arg1* edges using the *nsubj* and *dobj* edges respectively, indicating the arguments of the regulation: regulates (microRNA, metastasis).

```
########active verb form Phase 2 rule########
RuleID : openIE_active_1
Cond_0 : {tag:/VB.*/}=N0
Cond_1 : {}=N0 >/(nsubj|nsubj_null)/ {}=N1
Cond_2 : {}=N0 >dobj {}=N2
Action_1 : N0 >> arg0 >> N1
Action_2 : N0 >> arg1 >> N2
```

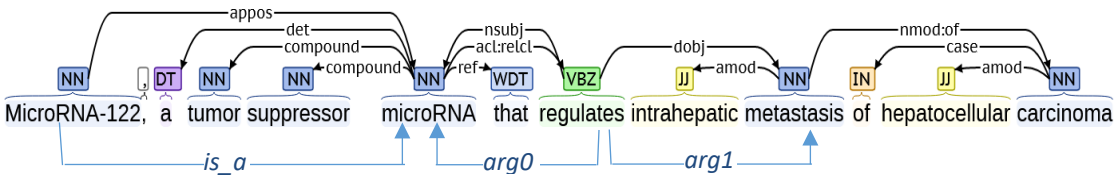


Figure E.4: Dependency Graph after applying Phase 2 set of rules

Recall as discussed in Chapter 2, we propagate the numbered argument edges based on "is-a" edges to extract the correct arguments of a trigger. Note, here the "arg0" edge from "regulates" should also point to "MicroRNA-122". This propagation of numbered-argument edges using extra-syntactic information is done using rules defined in Phase 3. The result of the application of the Phase 3 propagation rule (shown in the text box below) is depicted in Figure E.5. In this case, the *arg0* and *is_a* edges are combined and a new *arg0_prop* edge added, indicating the propagation of the *arg0* edge.

```
########Propagating Phase 3 rule########
RuleID : prop_isa_arg_1
Cond_0 : {}=N0 >/arg.*/=R1 {}=N1
Cond_1 : {}=N1 <is_a {}=N2
Action_1 : N0 >> R1_prop >> N2
```

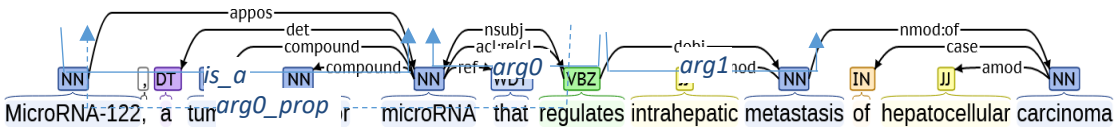


Figure E.5: Dependency Graph after applying Phase 3 set of rules: Propagation

Now that we have Arg0/1 edges pointing to the correct tokens in the sentences for the regulation relation, the next step involves extracting the argument phrases headed by these tokens. We specifically extract the predicate trigger, which is the governor token of the Arg0/1 edge, and the dependent token of the Arg0/1 edges, which are the arguments of the relation. We additionally extract base noun phrases (NP) headed by the Arg0/1 head noun and the full noun phrases with any prepositional attachments. The extraction of the base NP and the full NP is based on the constituency parse tree. Below are the different elements extracted by the framework:

a) **Trigger:** regulates
b) **Relation**: Regulation
c) **Arg0 head:** MicroRNA-122
d) **Arg 1 head:** metastasis

e) **Arg 0 base NP:** MicroRNA-122, a tumor suppressor microRNA

f) **Arg 1 base NP:** intrahepatic metastasis

g) **Arg 0 full NP:** MicroRNA-122, a tumor suppressor microRNA

h) **Arg 1 full NP:** intrahepatic metastasis of hepatocellular carcinoma

## Appendix F

## PERMISSIONS

Part of chapter 3 was published as an open-access article in "Journal of Biomedical Semantics (JBMS)". Their open-access policy follows the BMC license agreement, which is as follows:

"I, and all co-authors, agree that the article, if editorially accepted for publication, shall be licensed under the Creative Commons Attribution License 4.0. In line with BMC's Open Data Policy, data included in the article shall be made available under the Creative Commons 1.0 Public Domain Dedication waiver, unless otherwise stated. If the law requires that the article be published in the public domain, I/we will notify BMC at the time of submission, and in such cases not only the data but also the article shall be released under the Creative Commons 1.0 Public Domain Dedication waiver. For the avoidance of doubt it is stated that sections 1 and 2 of this license agreement shall apply and prevail regardless of whether the article is published under Creative Commons Attribution License 4.0 or the Creative Commons 1.0 Public Domain Dedication waiver."

(Source: https://www.biomedcentral.com/about/policies/license-agreement)

Part of chapter 6 was published as an open-access article in "Database: The Journal of Biological Databases and Curation". Their copyright policy is as follows:

"Database articles are published under the Creative Commons CC-BY licence. This means that users of Database articles are entitled to use, reproduce, disseminate or display these articles, including for commercial purposes provided that:

(1) the original authorship is properly and fully attributed;

(2) the journal and publisher are attributed as the original place of publication with correct citation details given;

(3) if an original work is subsequently reproduced or disseminated not in its entirety but only in part or as a derivative work this is clearly indicated;"

(Source: https://academic.oup.com/database/pages/About)