COMP90055 Computing Project Report

SurName: Teng

Given Name: Ruichen

Student Number: 678693

University Email: tengr@student.unimelb.edu.au

Name of Degree Enrolled in: Master of Information Technology — Computing

Subject Code: COMP90055 Computing Project

Total credit points for entire project: 25

Type of Project: Research project

Semester in which project commenced: Semester 1, 2015

Semester in which project is expected to complete: Semester1, 2015

Project Title: Information Extraction of biomedical relationships in published colon cancer literature

Supervisor: Prof. Karin Verspoor

Information Extraction of biomedical relationships in published colon cancer literature

Ruichen Teng

Department of Computing and Information Systems

University of Melbourne

This dissertation is submitted for the degree of

Master of Information Technology

June 2015

To my loving parents . . .

Declaration

I certify that

- this thesis does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any university; and that to the best of my knowledge and belief it does not contain any material previously published or written by another person where due reference is not made in the text.

- the thesis is approximately 6000 words in length (excluding text in images, table, bibliographies and appendices).

Ruichen Teng
June 2015

Acknowledgements

First and foremost, I want to thank my supervisor Prof. Karin Verspoor for the extremely helpful guidance and constant encouragement throughout the duration of this project. Her timely responses have really made this project work. In addition, my gratitude goes to Dr. Haibin Liu and Dr. Andrew MacKinlay for sharing valuable insights into the software system. I would also like to thank Dr. Jeremy Nicholson for chatting with me about NLP research. Finally I would like to thank my friends and family for carrying me through this extremely tough semester.

Abstract

Automatic information extraction from text discovers the underlying, structured information within unstructured text and has a wide variety of applications in this “big data” era. Particularly, relation extraction from biomedical literature can expand the biomedical knowledge base and has inspired a significant interest in research. A variety of techniques, including pattern-based methods, co-occurrence based methods, feature-based methods, semi-supervised methods and kernel methods have been proposed and evaluated on different relation extraction tasks. In this project, we adapted Approximate Subgraph Matching (ASM)[24], an existing event-extraction system to a new corpus — the Variome Corpus [40] for relation extraction tasks. We showed that merely changing the relation annotations to event annotations result in poor system performance. This is because fundamentally, trigger detection and prediction is an important process in event extraction, which however, is not included in relation extraction. Therefore, a full adaptation of the system is required to really explore the effectiveness of Approximate Subgraph Matching algorithms on relation extraction tasks.

Table of contents

List of figures

List of tables

Introduction

1.1 Motivation

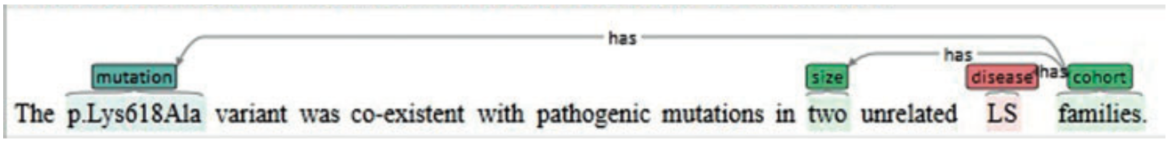
Text mining is the process of searching for patterns in natural language text using methods in computer science, linguistics, and statistics. Despite being unstructured and only human-understandable, text is still our primary media for exchange of information[43]. The prevalence of textual data presents a big challenge to computer-driven natural language understanding. *Information extraction*, in particular, refers to the task of acquiring organized, structured and queryable format of data from the unstructured corpus.

While text mining is widely used in areas like marketing and document verifying, it has received increased attention for its application to biomedical literatures[3, 20, 22]. This trend stems from the direct need of biomedical workers and researchers to cope with information explosion in their field. For instance, MEDLINE(Medical Literature Analysis and Retrieval System Online), the online database of United States National Library of Medicine, has accumulated nearly 0.8 million citations and 2.7 billion searches in 2014 alone[27], with total citations reaching 25 million. Within these publications there are valuable research results that should add to human knowledge. In the meantime, our primary knowledge base in life science - the biomedical databases, are still mostly being populated manually by *biocurators* - the “museum catalogers of the Internet age”[41]. They are professional scientists who read biomedical articles, record relevant data and organize them according to the biomedical database schema. The sheer volume of publications has made this process increasingly unrealistic[11].

Not only does data overload make knowledge discovery demanding, it also leads to a decline in literature quality. Nowadays biomedical workers and researchers are more prone to drawing wrong conclusions because they simply can not read all the relevant publications, among which oftentimes contradicting results are reported. Needless to say, we are in desperate need of automatic tools for systematically analyzing documents and extracting information. In fact, it has been argued that text mining is required to improve the coverage of databases[4].

1.2 Definitions and Assumptions

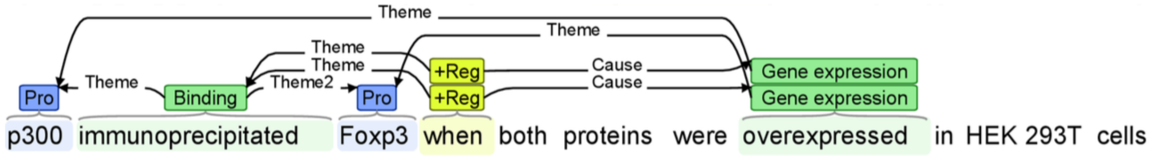
Below are a list of terms which I will use throughout this thesis, more detailed examples will be given in the relevant sections, but a general definition is first given here to avoid any confusions that might arise.



The p.Lys618Ala variant was co-existent with pathogenic mutations in two unrelated LS families.

Fig. 1.1 An Example of Relations Captured in a Sentence

- *Relation Extraction:* In general, relation extraction refers to the process of discovering a relationship between entities in text. Relations can be unary (one-to-itself), binary (one-to-one) and more complex[26]. In this project we are mainly concerned with binary relations between two entities. As illustrated in Figure 1.1, the sentence contains relations like *families have p.Lys618A1a*, *families have two(members)*, *families have the LS(disease)*, etc. In the domain of natural language processing, the relation can be semantic or syntactic, with semantic relations being the most important for knowledge discovery. Specifically, semantic relations between biomedical entities such as protein-protein interactions and gene expressions cover a wide range of knowledge in this field and have driven numerous research work.



p300 immunoprecipitated Foxp3 when both proteins were overexpressed in HEK 293T cells

Fig. 1.2 An Example of Events Captured in a Sentence

- *Event Extraction:* A biomedical event usually refers to a statement of molecular interaction in text [6]. Figure 1.2 shows a sentence that contains two protein catabolism events (pro), one binding event, two positive regulation events, and two gene expression events between respective entities.

Both relation extraction and event extraction belongs to the broader concept of information extraction.

1.3 Research Question

This project looks to investigate the adaptation of event extraction system Approximate Subgraph Matching (ASM)[24], on relation extraction tasks, specifically regarding the relations in the Variome Corpus annotated with the Variome Annotation Schema[40]. The effectiveness of the system in extracting relations between entities (disease, ethnicity, gender, gene, mutation, patient, etc.) will be evaluated. Since ASM is a supervised, rule-based data mining system developed solely for event extraction, it needs to be re-trained and carefully adapted to the new corpus. We believe such a relation extraction tool has very promising applications for biomedical researchers, doctors, pharmaceutical companies, and the general public. The extracted relations can be helpful in information search, knowledge discovery and hypothesis generation.

1.4 Thesis Structure

The remainder of the thesis is organized in the following manner. In Chapter 2, we discuss different algorithms used for relation extraction tasks including pattern-based methods, co-occurrence based methods, feature-based methods, semi-supervised methods and kernel based methods. In Chapter 3, we provide a detailed description of the Approximate Subgraph Matching system and explain how this event extraction system was adapted for relation extraction task in Variome Corpus. We then present the relation extraction results and our analysis. In Chapter 4, we conclude our work and present future directions.

Related Work

This chapter will explore the research related to this thesis. First we introduce briefly named entity recognition, an important step before relation extraction. Next different relation extraction algorithms are presented.

2.1 Named Entity Recognition

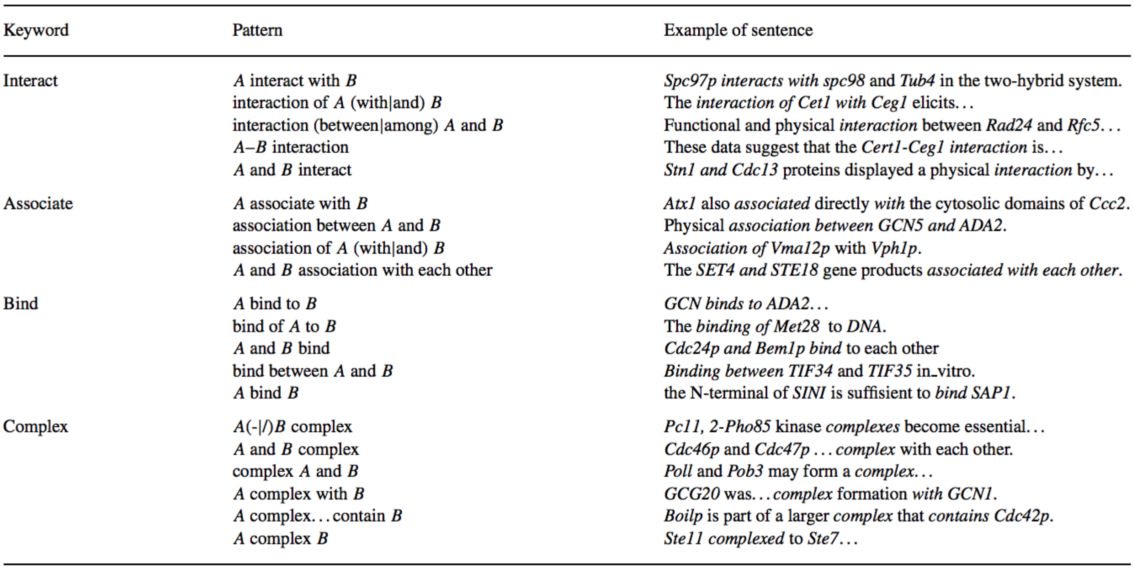
Named Entity Recognition, or NER, is the task of identifying elements in text that belong to pre-defined categories. Specifically, NER in biomedical text mining aims at identifying entities such as proteins, diseases, genes, etc. There is extra difficulty for NER in the biomedical domain mainly for the following reasons. First, there is no canonical dictionary for biomedical entities[11]. The entity names are created on the fly and the number of names is in the millions. Second, the names are not defined unambiguously[42]. The same name may refer to different entities depending on context, in the meantime an entity might have several names. Finally, even human interpretations differ with named entities in the biomedical text. For example in our corpus for this project there were about 10% disagreement on named entities between two human annotators[40].

2.2 Relation Extraction

While it is often the case that the accuracy of named entity recognition would have a significant impact on the performance of relation extraction, the two problems are usually treated separately. This allows the relevant tools for each problem to be evaluated independently. Therefore, in the following sections, the relation extraction methods usually have the named entity annotations given in their test data, so that the algorithm only needs to focus on extracting relations between known entities.

2.2.1 Pattern Based Methods

Pattern based relation extraction methods first saw its application in extracting protein-protein interactions[16]. Initially, a set of part-of-speech rules are applied to split the syntactically complex sentence into simple sentences. For example, a sentence with part-of-speech tag sequence $\{P1,\ VB1,\ P2,\ VB2,\ CC,\ P3\}$ can be splitted to $\{P1,\ VB1,\ P2\}$ and $\{P1,\ VB2,\ P3\}$. Then, a set of word patterns can be applied to extract relations from these simple sentences[31]. An example of that is shown in Figure 2.1. In addition, Hao et al. [16] proposed an idea of minimal description length, as in finding the optimal pattern set that has the most balance among high precision, short rule length and low rule complexity by dynamic programming. Pattern based methods has achieved quite respectable performances for extracting protein-protein interactions[16]. However, it does not incorporate the richness of expressions, such as the anaphora terms like pronouns. Consequently, it does not generalize well and usually needs huge amount of training data.



Keyword	Pattern	Example of sentence
Interact	*A* interact with *B*	*Spc97p interacts with spc98* and *Tub4* in the two-hybrid system.
	interaction of *A* (with|and) *B*	The *interaction of Cet1 with Ceg1* elicits…
	interaction (between|among) *A* and *B*	Functional and physical *interaction* between *Rad24* and *Rfc5*…
	A–*B* interaction	These data suggest that the *Cert1-Ceg1 interaction* is…
	A and *B* interact	*Stn1 and Cdc13* proteins displayed a physical *interaction* by…
Associate	*A* associate with *B*	*Atx1* also *associated* directly *with* the cytosolic domains of *Ccc2*.
	association between *A* and *B*	Physical *association between GCN5 and ADA2*.
	association of *A* (with|and) *B*	*Association of Vma12p* with *Vph1p*.
	A and *B* association with each other	The *SET4 and STE18* gene products *associated with each other*.
Bind	*A* bind to *B*	*GCN binds to ADA2*…
	bind of *A* to *B*	The *binding of Met28* to *DNA*.
	A and *B* bind	*Cdc24p and Bem1p bind* to each other
	bind between *A* and *B*	*Binding between TIF34* and *TIF35* in_vitro.
	A bind *B*	the N-terminal of *SINI* is suffisient to *bind SAP1*.
Complex	*A*(-|/)*B* complex	*Pc11, 2-Pho85* kinase *complexes* become essential…
	A and *B* complex	*Cdc46p* and *Cdc47p* … *complex* with each other.
	complex *A* and *B*	*Poll* and *Pob3* may form a *complex*…
	A complex with *B*	*GCG20* was… *complex* formation *with GCN1*.
	A complex… contain *B*	*Boilp* is part of a larger *complex* that *contains Cdc42p*.
	A complex *B*	*Ste11 complexed* to *Ste7*…

Fig. 2.1 Word Patterns for Protein-Protein Interaction from Ono et al. [31]

2.2.2 Co-occurrence based methods

Finding co-occurring terms within a sentence or abstract has been the foundation of many relation extraction algorithms[5, 9, 17, 35, 36, 44]. Simple co-occurrence measures include probabilistic indicators such as point-wise mutual information, chi-square and log-likelihood ratio. More advanced measures include Concept Space, where thesaurus are mapped to a multi-dimensional Euclidean space[23, 38]. The main advantage of co-occurrence based methods is their simple implementation and low computational complexity. However, simplistic word counting often fails to grasp the essence of relations. Thus co-occurrence based methods are more suitable for detecting simple relations like gene-gene relations, but not in the general case of more complex biological relations.

2.2.3 Feature Based Methods

Feature based relation extraction often takes a statistical machine learning approach. The relation extraction task is regarded as a classification problem where each sentence with entities of interest is classified to a relation type. As a result, it is necessary to engineer a set of features for the learning algorithm. Kambhatla [19] from the IBM Watson lab proposed an approach that selects a feature stream between every pair of entity mentions. The feature stream include in-between word sequence, entity type, overlap with other mentions, syntactically dependent words of the mentions and parse-tree paths connecting two mentions. This feature stream is then used to build a maximum entropy model to classify relations for the Automatic Content Extraction(ACE) task. GuoDong et al. [15], Jiang and Zhai [18] later extended this work experimenting with more features. While their work showed very good results, they also rely heavily on high-quality feature engineering. Inclusion of undesirable features would significantly hurt system performance. This burden of feature engineering makes feature based methods less desirable[24].

2.2.4 Semi-Supervised Methods

Semi-supervised methods address the scarcity of training data. Training data for relation extraction is extremely expensive because substantial human labor is required to read the documents and label each training instance. Craven et al. [13] first experimented distant supervision by automatically extracting relations from databases records, "weakly" labeling the training data with these relations, extracting patterns from the training data, and filtering out inconfident patterns. Similar approaches have been explored since [7, 28, 29]. Recently Ravikumar et al. [32] used protein structure records in the Protein Data Bank for automatic creation of training data in protein-residue relation extraction. The problem with semi-supervised methods is that they could generate noisy patterns[30].

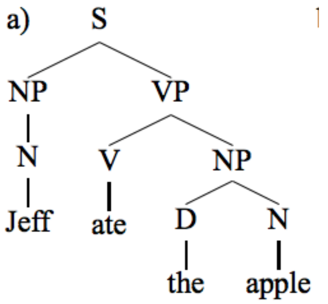


Fig. 2.2 Example of a Shallow Parse Tree [45]

2.2.5 Kernel Methods

Kernels provide a similarity measure between two objects in some complex feature space. In contrast to feature based methods, kernel-based methods allow the original representation of the object to be retained, and the kernel function will work out the similarity measure. For instance, the sentence *"Jeff ate the apple."* can represented as a shallow parse tree[45]. The feature based methods would want to select features such as number of nodes, number of edges and directions of edges, whereas kernel based methods allows feeding the entire tree object into a tree kernel function K[12], and output the similarity measure between two trees t_1 and t_2 as $K(t_1,t_2)$. Among various methods discussed above, we feel that **kernel based methods** can preserve the linguistically rich representations of sentences and has more flexibility as there are no manually encoded rules. The next chapter will discuss a graph kernel we chose for the relation extraction task in this project.

Project Work

3.1 Data Collection

Our dataset is the Variome Corpus[40], which is openly accessible. [1] Verspoor et al. [40] gave a detailed illustration of the document selection and annotation process. I will summarize the main points here.

A major part of the current biomedical research lies in understanding the relations between human genetic variation and disease phenotypes. The *Human Variome Project*, or *HVP*, is a global initiative to collect all genetic variation information affecting human health[33]. In particular, it acts as a liaison between individuals and organizations to integrate the genetic variants into databases that are open to the general public[40]. The *International Society for Gastrointestinal Hereditary Tumours (InSiGHT)*, is an international organization which aims to benefit patients with hereditary gastrointestinal(GI) tumours by research, education and personal assistance. In 2008, InSiGHT and HVP began a collaboration which propels InSiGHT to refine its process in the integration and interpretation of genetic variants. Consequently, a substantial effort was made to understand the mutation of mismatch repair(MMR) genes, the cause of Lynch Syndrome - one of the main syndromes of GI cancer[34]. A total of 10 full-text articles were selected from PubMed Central®by searching the common Lynch syndrome genes. These documents cover inherited colon cancer as well as certain other cancers. The annotation schema, also known as the Variome Annotation Schema[40], include 11 entity types and 13 relation types, as can be seen in the table 3.1.

In short, the annotation and selection of corpus is inspired by the needs of inSIGHT database, but intended for broader applications. In particular, the annotations are done by two human annotators. As suggested in [40], occasional annotation disagreement has been

[1] http://www.opennicta.com.au/home/health/variome

Table 3.1 Variome Relation Types

Relation Type	Entity1	Entity2
relatedTo	mutation	disease
relatedTo	disease	gene
relatedTo	disease	body-part
has	gene	mutation
has	mutation	size
has	disease	characteristic
has	cohort-patient	age
has	cohort-patient	characteristic
has	cohort-patient	disease
has	cohort-patient	ethnicity
has	cohort-patient	gender
has	cohort-patient	mutation
has	cohort-patient	size

resolved and the result is merged into a single corpus. Thus in this relation extraction task here, we refer to the manual annotations as the gold standard.

3.2 Dependency Graph and Shortest Path

The dependency graph of a sentence is a directed graph, where nodes represent sentence tokens, and edges indicate their semantic relations. Figure 3.1 shows the dependency graph of the sentence *" The p.Lys618Ala variant was co-existent with pathogenic mutations in two unrelated LS families."* generated by the Stanford Parser. Such a graph preserves the rich semantic structure of a sentence, and has been widely regarded as an informative way of presenting a sentence. A detailed explanation of the relative constituents in the graph can be found in [14]. However, the point is to transfer only human-readable sentences to a computer-understandable data structure. The general idea would be to feed this graph into a learning algorithm and classify relations based on similarity to the sentence graph in the training set. Different approaches exist for this process. Turku Event Extraction System (TEES)[2], for instance, engineers a feature vector which consists of token features(part-of-speech tags and character constituents for each word), sentence features(bag-of-words counts), and graph-based features(dependency path represented as N-grams) and builds a multi-class SVM model with the feature vector[6]. In this project, we decide to use the *Shortest Path Hypothesis*[8], namely the heuristic that the relation between two entities in

[2] https://github.com/jbjorne/TEES

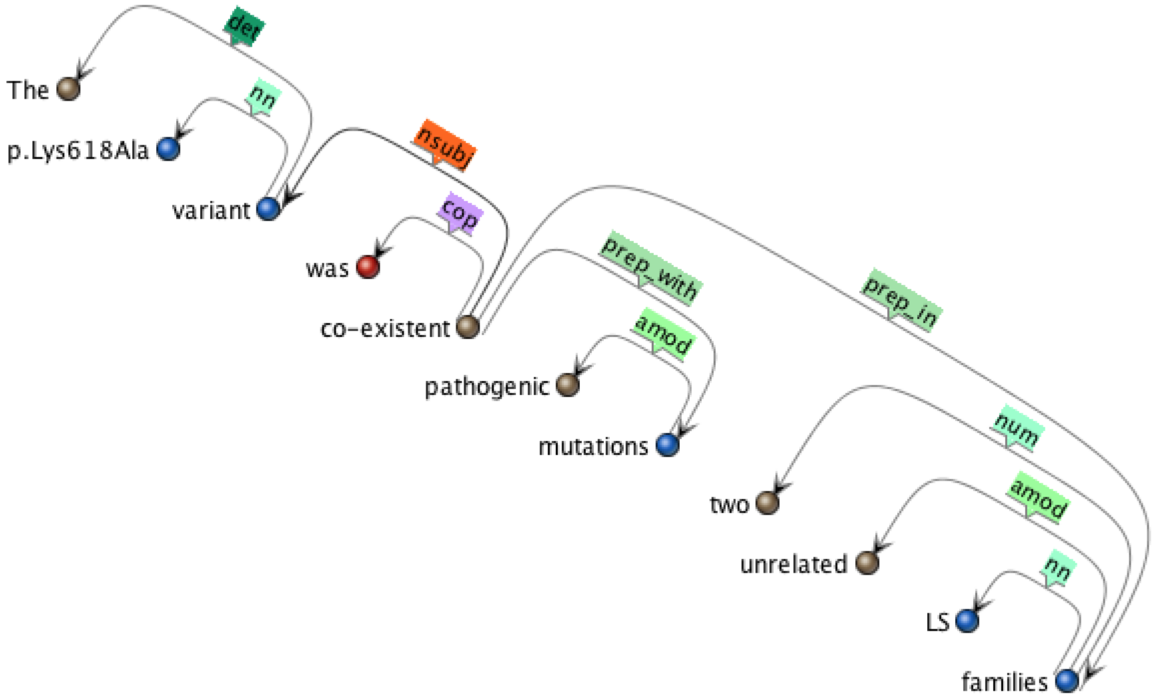


Fig. 3.1 Dependency Graph of " The p.Lys618Ala variant was co-existent with pathogenic mutations in two unrelated LS families."

a sentence can be distilled from the shortest path between these entities in the undirected version of the sentence dependency graph. This effectively reduces the burden of feature engineering[24], but it also calls for high-quality training data. We believe that with effective parameter tuning and clever graph matching techniques, the shortest path can be a single standalone feature for a relation between two entities.

3.3 Approximate Subgraph Matching Algorithm

Disclaimer: As the Approximate Subgraph Matching system was originally developed for event extraction, it is more convenient to refer to the system as being learning "events". By nature events are nothing more than complex relations between entities, which is exactly the rationale behind adapting such an event extraction system for relations extraction tasks. What was later done in the adaptation process was treating relation as a type of "event". In this section, "relations" and "events" are used (somewhat) interchangeably.

The Approximate Subgraph Matching framework, proposed by Liu et al. [24] has the following work flow:

3.3.1 Prepossessing

As mentioned in section 2.1, in order to separate Named Entity Recognition from the Relation Extraction task, the named entity annotations are provided in training, development and test sets. The sentences are identified by the JULIE Sentence Boundary Detector[37], and parsed with the *clearnlp* parser [3][10]. The model for dependency parsing and part-of-speech tagging is trained on the CRAFT corpus [39].

3.3.2 Rule Learning

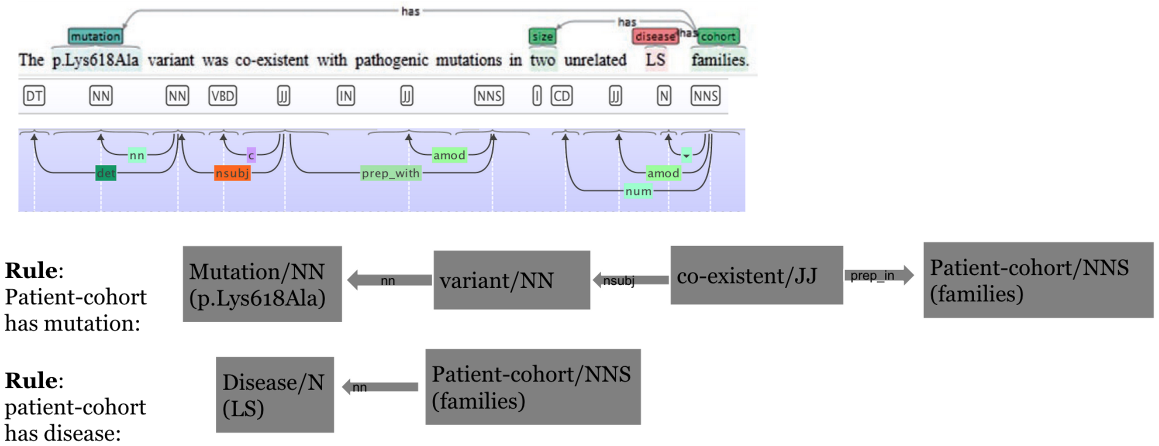

has
has
mutation
size
disease
has
cohort
The p.Lys618Ala variant was co-existent with pathogenic mutations in two unrelated LS families.
DT NN NN VBD JJ IN JJ NNS I CD JJ N NNS
nn
det
nsubj
prep_with
amod
amod
num
Rule: Patient-cohort has mutation:
Mutation/NN (p.Lys618Ala)
nn
variant/NN
nsubj
co-existent/JJ
prep_in
Patient-cohort/NNS (families)
Rule: patient-cohort has disease:
Disease/N (LS)
nn
Patient-cohort/NNS (families)

Fig. 3.2 Rules learned for “ The p.Lys618Ala variant was co-existent with pathogenic mutations in two unrelated LS families.”

For each annotated sentence in the training set, a dependency graph is generated and the nodes representing entities are marked. The entity tokens are then replaced with their entity types, such that the rules learned are about an the generic entity types (e.g. gene mutation) instead of specific entities (e.g. p.Lys618Ala), so that our model has a better ability to generalize as we might not see the specific entities again in the test set. Next, the graph is transformed to its undirected version and the shortest path between entities are found with Dijkstra algorithm. This path is leaned as a rule associated with the event type in the annotation as a rule corresponding to that event type. Figure 3.2 gives an example of

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

the learned rules for sentence *“ The p.Lys618Ala variant was co-existent with pathogenic mutations in two unrelated LS families.”* This kind of instance-based learning can be effective provided there is enough training data[2]. A set of rules would then be learned for each event type, representing the graph patterns that indicating a specific type of event.

3.3.3 Sentence Matching

The rules generated from the previous step could, in theory, be used to match sentences. For each sentence in the test set, a dependency graph can be generated together with the entity annotations(these are provided, as discussed). The graph can be searched exhaustively looking at the path(s) between each entity tokens, for an exact match with one or more rules within the rule set. This step is know as searching for exact subgraph isomorphism. Figure 3.3 gives an example of Exact Subgraph Matching(ESM).

However, the above-mentioned matching approach would invariably lead to low recall, as the richness of language can almost always produce slightly different dependency graph structure representing exactly the same events between the same entities. For instance, Figure 3.4 shows a scenario where *patient-cohort has mutation* should be extracted as an event of interest, but there is a slight mismatch in the subgraph patterns. This leads to the rationale behind approximate subgraph matching, which relaxes the matching process and allows for a

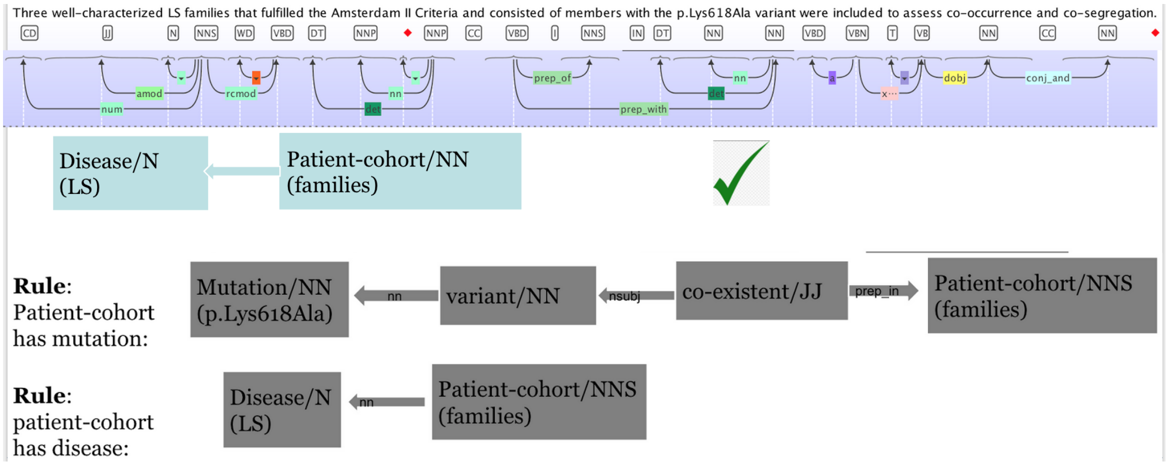
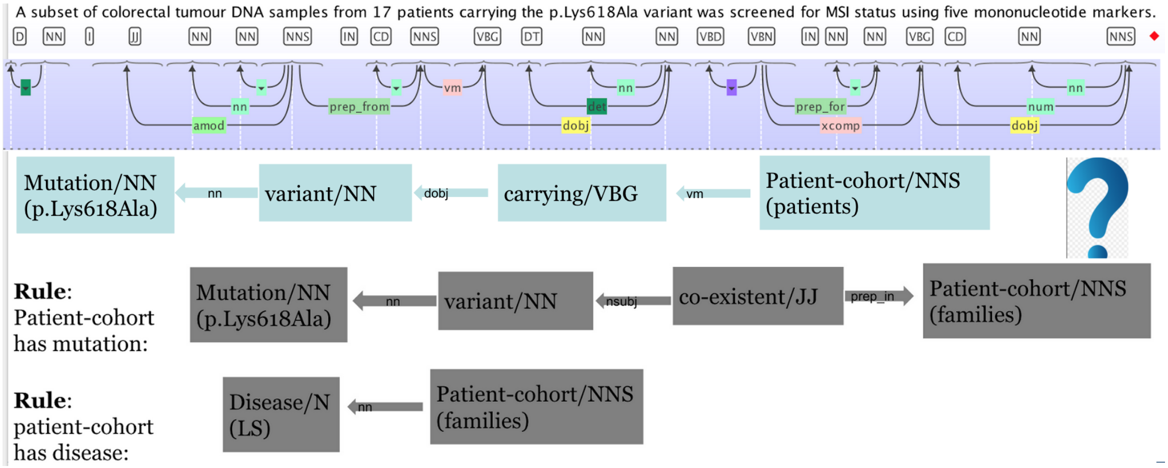

Three well-characterized LS families that fulfilled the Amsterdam II Criteria and consisted of members with the p.Lys618Ala variant were included to assess co-occurrence and co-segregation.
CD JJ N NNS WD VBD DT NNP NNP CC VBD I NNS IN DT NN NN VBD VBN T VB NN CC NN
num
amod
rcmod
det
nn
prep_of
prep_with
det
nn
a
x…
dobj
conj_and
Disease/N (LS)
Patient-cohort/NN (families)
✓
Rule: Patient-cohort has mutation:
Mutation/NN (p.Lys618Ala)
nn
variant/NN
nsubj
co-existent/JJ
prep_in
Patient-cohort/NNS (families)
Rule: patient-cohort has disease:
Disease/N (LS)
nn
Patient-cohort/NNS (families)

Fig. 3.3 Exact Subgraph Matching



A subset of colorectal tumour DNA samples from 17 patients carrying the p.Lys618Ala variant was screened for MSI status using five mononucleotide markers.

D NN J JJ NN NN NNS IN CD NNS VBG DT NN NN VBD VBN IN NN NN VBG CD NN NNS
amod
nn
prep_from
vm
det
dobj
nn
prep_for
xcomp
num
dobj
Mutation/NN (p.Lys618Ala)
nn
variant/NN
dobj
carrying/VBG
vm
Patient-cohort/NNS (patients)
?
Rule: Patient-cohort has mutation:
Mutation/NN (p.Lys618Ala)
nn
variant/NN
nsubj
co-existent/JJ
prep_in
Patient-cohort/NNS (families)
Rule: patient-cohort has disease:
Disease/N (LS)
nn
Patient-cohort/NNS (families)

Fig. 3.4 Approximate Subgraph Matching

penalty-based matching. The formula for calculating subgraph distance is:

$$\begin{aligned} GraphDist(SentenceGraph, RuleGraph) = \\ w1 \times structDist(SentenceGraph, RuleGraph) + \\ w2 \times labelDist(SentenceGraph, RuleGraph) + \\ w3 \times directionalityDist(SentenceGraph, RuleGraph) \end{aligned} \tag{3.1}$$

structDist is the structural difference between two subgraphs denoted by the distance between each pair of matched nodes. *labelDist* and *directionalityDist* are the difference in edge labels and edge directions respectively[24]. A different distance threshold t, is set for each event type such that if the subgraph distance calculated according to 3.1 is smaller than t, the subgraph is considered a match to the rule graph and the entity tokens are considered as having a relation of the type which the rule graph belongs to. Otherwise the sentence is counted as a mismatch.

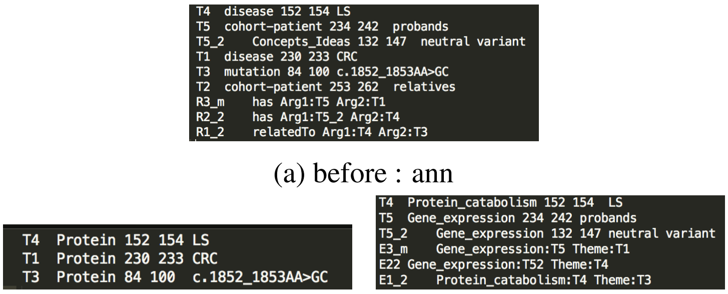
3.3.4 Rule Optimization

Finally, to avoid learning the idiosyncrasies in the training data, an iterative rule set optimization process is executed. After the initial learning phase, each rule in the rule set is tested on the training data first to see if it get produce accurate enough predictions (above 0.25). The low performing rules are discarded as a consequence.

3.4 System Adaptation

As mentioned in 1.3, the project aims at adapting an existing event extraction system on relation extraction tasks. The current ASM system was developed for the GENIA Event (GE) task of BioNLP Shared Task 2011 and 2013.

BioNLP shared Task series is a community-wide text mining challenge specifically for biomedical literatures. The *Genia Event Extraction for NFkB knowledge base construction* (GE) task in BioNLP shared task 2013 aims at retrieving biological events such as gene expression, protein catabolism, protein binding and localization, etc. [4] The annotations are of the following format: The *.a1* files list all the entities as shown in Figure 3.5b, whereas the *.a2* files list all the event triggers, followed by events as shown in Figure 3.5c. An entity is annotated with its ID *(T1)*, entity type *(Protein)*, text offsets *(230-233)*, and the textual token *(CRC)*. Similarly, a trigger is annotated with its ID *(T5)*, event type *(Gene Expression)*, text offsets *(234-242)*, and the textual token *(probands)*. A trigger is usually a verb that "triggers" an event, but can be of other part-of-speech constituents. An event is annotated with an event ID ID *(E3_m)*, event type *(Gene Expression)*, event trigger ID *(T5)*, and event theme ID *(T1)*. An event usually indicates a relationship between the event trigger and one or more entities.



```
T4  disease 152 154 LS
T5  cohort-patient 234 242  probands
T5_2    Concepts_Ideas 132 147  neutral variant
T1  disease 230 233 CRC
T3  mutation 84 100 c.1852_1853AA>GC
T2  cohort-patient 253 262  relatives
R3_m    has Arg1:T5 Arg2:T1
R2_2    has Arg1:T5_2 Arg2:T4
R1_2    relatedTo Arg1:T4 Arg2:T3
```

(a) before : ann

```
T4  Protein 152 154 LS
T1  Protein 230 233 CRC
T3  Protein 84 100  c.1852_1853AA>GC
```

(b) after: a1

```
T4  Protein_catabolism 152 154  LS
T5  Gene_expression 234 242 probands
T5_2    Gene_expression 132 147 neutral variant
E3_m    Gene_expression:T5 Theme:T1
E22 Gene_expression:T52 Theme:T4
E1_2    Protein_catabolism:T4 Theme:T3
```

(c) after: a2

Fig. 3.5 Changing Annotation Format

Figure 3.5a shows the *Variome Annotation Format*. Different from the shared task, all the annotations would be in one *.ann* file, with entities annotated the same way as in the shared task, and relations similar to events. However, as can be seen in Figure 3.5a, relations do not have triggers at all. This distinction became a major challenge for this project as the existing ASM system has intrinsic expectation of triggers across the entire system. In the duration

[4] A detailed event list can be found here: http://bionlp.dbcls.jp/projects/bionlp-st-ge-2013/wiki/IEevaluation

of this project, all of my attempts to fully adapt the system (change the code) to relation extraction tasks have failed.

Not able to fully change the ASM system code, I decided to transform the annotation format of the Variome Corpus to be the BioNLP Shared Task format, such that the ASM system would not break. The transformation is illustrated in Figure 3.5. The “has” relationship in the annotation is changed to be a “gene regulation”event, and the “relatedTo” relationship is changed to be a “Phosphyrilation” event. The entities in the Variome Annotation would all be changed to type “Protein”. **The major bottleneck for this adaptation work is that the original system includes a hard-coded trigger detection and prediction process.** Before extracting events, each word in the test sentence is given trigger score based on the trigger classifier (multi-class SVM) learned from training data. An event trigger is predicted before the subgraph connecting trigger node and entity nodes is matched against the rule graph. Detecting trigger words such as “activates” or a gene expression event is an important step for event extraction. However, this process is not included at all in the relation extraction task. To cope with the original system setup, I had to add some “fake triggers” for the relations masked as events. Therefore, the first entity argument of each relation annotation is added as the trigger in order for the relation to be regarded as a valid event after changing its annotation, as illustrated in Figure 3.5c.

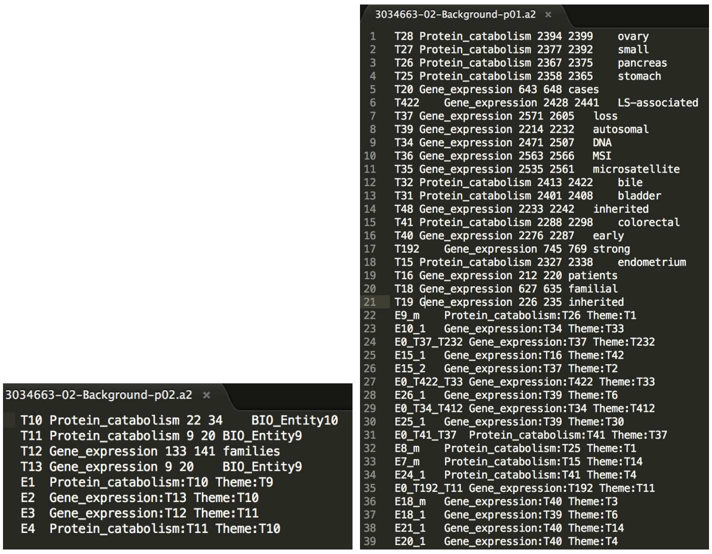
To split the original Variome Annotation file (.ann) into two files (.a1 and .a2), I wrote code to scan through the entities and relations in the .ann file. For each entity, my code checks first if the entity exists in any relation annotations. If not, the annotation will be ignored, e.g. $T2$. Next the code checks if the entity is the first argument or the second argument of a relation annotation. If the entity is the first argument, it is treated as an event trigger, such as $T4, T5, T5\_2$ and put in the .a2 file, otherwise it is put in the .a1 file.

3.5 Results and Analysis

The overall result of the system is shown in table 3.2. The main reason the system is not performing well is that it is not predicting the triggers words correctly. Take look at an example comparison between a predicted .a2 file in Figure 3.6a and the gold standard .a2 file in Figure 3.6b, almost all the trigger predictions are incorrect. As mentioned before, trigger detection is treated as a classification problem by the system and the detected trigger is later included as a node in the subgraph matching process. Because I have chosen to treat entity annotations

Table 3.2 Overall Result

Relation Type	Gold	Answer	Match	Precision	Recall	F1-score
has	1711	1310	402	0.3069	0.2350	0.2661
relatedTo	157	1498	36	0.0240	0.2293	0.0435
TOTAL	1868	2808	438	0.1560	0.2345	0.1874



3034663-02-Background-p02.a2

```
T10 Protein_catabolism 22 34    BIO_Entity10
T11 Protein_catabolism 9 20 BIO_Entity9
T12 Gene_expression 133 141 families
T13 Gene_expression 9 20    BIO_Entity9
E1  Protein_catabolism:T10 Theme:T9
E2  Gene_expression:T13 Theme:T10
E3  Gene_expression:T12 Theme:T11
E4  Protein_catabolism:T11 Theme:T10
```

3034663-02-Background-p01.a2

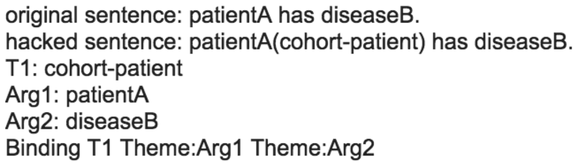
```
T28 Protein_catabolism 2394 2399    ovary
T27 Protein_catabolism 2377 2392    small
T26 Protein_catabolism 2367 2375    pancreas
T25 Protein_catabolism 2358 2365    stomach
T20 Gene_expression 643 648 cases
T422    Gene_expression 2428 2441   LS-associated
T37 Gene_expression 2571 2605   loss
T39 Gene_expression 2214 2232   autosomal
T34 Gene_expression 2471 2507   DNA
T36 Gene_expression 2563 2566   MSI
T35 Gene_expression 2535 2561   microsatellite
T32 Protein_catabolism 2413 2422    bile
T31 Protein_catabolism 2401 2408    bladder
T48 Gene_expression 2233 2242   inherited
T41 Protein_catabolism 2288 2298    colorectal
T40 Gene_expression 2276 2287   early
T192    Gene_expression 745 769 strong
T15 Protein_catabolism 2327 2338    endometrium
T16 Gene_expression 212 220 patients
T18 Gene_expression 627 635 familial
T19 Gene_expression 226 235 inherited
E9_m    Protein_catabolism:T26 Theme:T1
E10_1   Gene_expression:T34 Theme:T33
E0_T37_T232 Gene_expression:T37 Theme:T232
E15_1   Gene_expression:T16 Theme:T42
E15_2   Gene_expression:T37 Theme:T2
E0_T422_T33 Gene_expression:T422 Theme:T33
E26_1   Gene_expression:T39 Theme:T6
E0_T34_T412 Gene_expression:T34 Theme:T412
E25_1   Gene_expression:T39 Theme:T30
E0_T41_T37  Protein_catabolism:T41 Theme:T37
E8_m    Protein_catabolism:T25 Theme:T1
E7_m    Protein_catabolism:T15 Theme:T14
E24_1   Protein_catabolism:T41 Theme:T4
E0_T192_T11 Gene_expression:T192 Theme:T11
E18_m   Gene_expression:T40 Theme:T3
E18_1   Gene_expression:T39 Theme:T6
E21_1   Gene_expression:T40 Theme:T14
E20_1   Gene_expression:T40 Theme:T4
```

(a) Prediction (b) Answer

Fig. 3.6 Prediction vs. Answer

as triggers for the relations masked as events, the system would predict the triggers based on what it has learned about the triggers and the event(relation) type they corresponds to in the training data. Because the triggers are "fake triggers", there exist far less deterministic relationship between "fake triggers" and the event(relation) types than a real trigger, e.g. "activates" for the event type "gene expression". Indeed, an entity can be involved in many event(relation) types, and might as well be evenly distributed among these event(relation) types. The trigger then would not be favored against any specific event(relation) type. It can be imagined that a trigger classifier built by this kind of training data would make every poor predictions.

After this attempt, I have had a few better ideas to add "fake triggers" which I did not have time to implement, the best one being adding the entity types (patient-cohort), with parenthesis directly after the actual entity tokens in the sentence, and change the relation annotation to a binding event annotation. For instance as in Figure 3.7. This step should be done for both the training and test data. That way it alleviates the trigger prediction problem



```
original sentence: patientA has diseaseB.
hacked sentence: patientA(cohort-patient) has diseaseB.
T1: cohort-patient
Arg1: patientA
Arg2: diseaseB
Binding T1 Theme:Arg1 Theme:Arg2
```

Fig. 3.7 Adding Fake Triggers

as all relations masked as events would have some certain entity type as triggers, such that the trigger prediction accuracy would improve. However there still exists a problem because in the test data there is no way to know which entity would be a the first argument such that the trigger can be added to.

Conclusion

4.1 Conclusion

We have explored the possibility of adapting an event-extraction system in relation extraction tasks for the Variome Corpus.

4.2 Contributions

The contributions of the research in this thesis are as follows:

- *The first attempt to use the system outside original system developers.* While this may seem like an easy task, a lot of the time for this project has been devoted to resolving dependencies, and debugging scripts to first have an end-to-end system, even for the original shared task data set.
- *The first attempt for literature mining of the Variome Corpus.* To our knowledge, this is the first text mining attempt to the Variome Corpus.

4.3 Future Work

4.3.1 Full System Adaptation

Given the nature of the existing software system, in effect I wrote an adapter for the Variome Corpus, transforming the Variome Corpus Annotations to a format that the ASM system can work with. This is, of course, far from the best solution. However, the ASM system was originally developed solely for an event extraction shared task under time constraints, without much expectation that the system might be adapted on day. As a result, the system annotation format hard-coded in strings and a lot of implicit programming logic such as the involvement of trigger spread out across many different methods, to the extent that all of my efforts to change the code have failed. The main reason for the failure is my inability to detect all the methods and classes where changes should be made, resulting in unexpected system behavior. While is natural and understandable for academic software like this to be one-and-done after use in the shared task, I think it is in the interest of future system users and developers to have a well-documented, adequately extensible system in place, with room for tweaking algorithms and file formats with parameters. Ideally, the shortest path kernel will expose interfaces to users, allowing them to adjust what is happening under the hood with a few parameters. For instance, the existence of trigger should be optional and can be set with a parameter applied to the both the rule learning process and event extraction process. In that way, when learning relations instead of events, triggers can be effectively ignored. In fact, even in the shared task 2013, the existence of trigger for relations and coherence is optional. Next, annotation format should be parametrized, and the system should establish a relationship between user-defined entity and relation types its the entity and relation class. Moreover, the named entity recognition could be a dispatchable unit of the system too, with options to use user-given entities or the output of a named-entity recognizer. A perfect scenario would be one in which the user can input a configuration schema indicating annotation formats, entity types, entity output(whether from the annotation file or from a named entity recognizer), event/relations types and the location of textual data. That way the system can be used for different kinds of tasks, possibly even beyond the domain of biomedical text mining and provide valuable feedback for the plausibility of the approximate subgraph matching paradigm.

4.3.2 Relation Type Fine-Tuning

Currently, the system only has two relations types, a “has” relation and a “relatedTo” relation. However, a patient-has-disease relation is semantically quite different from a gene-has-mutation relation, despite both of them being chunked to a single relation type called “has”. One might be tempted to have fine-grained relation types and incorporate the entity types into the relation, such as a separate “patient-cohort-has-disease” and a separate “gene-has-mutation” instead of a “has” relation for both. Nevertheless, this would cause the training set to become extremely sparse and the system learning behavior could change drastically. The correlation between granularity of event types and system performance has yet to be explored.

4.3.3 The Parser Effect

This project uses the exact same preprocessing pipeline as [25]. However, MacKinlay et al. [25] concluded that changing the parser from the one originally used in [24] has limited recall to an effect that can not be offset by increasing the amount of training data. It was suggested that the longer dependency graph produced by the *clearnlp* parser is harder to generalize. The effect of parser on relation extraction task for the Variome Corpus is yet to be investigated. Again, the parser should be a loosely coupled component of the system too, such that the effect of different dependency parsers can be explored more easily.

4.3.4 Parameter Tuning

There are a number of different parameters in the system. There is the weight distribution across different components (graph distance, structure distance, label distance) in calculating subgraph distance; There is the distance thresholds in deciding whether a subgraph is an approximate match to the rule graph; There is the precision requirement (currently set as 0.25) in the rule set optimization process. All of these parameters are currently set to be the same as in [25] for the BioNLP shared task 2013. However, these parameters are likely to be corpus and task dependent and need to be tuned for the Variome Corpus on the relation extraction task.

4.3.5 The Abstract Effect

Many earlier biomedical text mining approaches only process abstracts of articles. The rationale is that abstract would contain a summary of the whole article and the important relations that it contains. In the Variome Corpus the full-text articles are splitted into sections such as abstract, introduction, conclusion, etc. With this in mind, one might be careful in how to distribute data for training, development and testing to avoid over-fitting in the future.

4.3.6 Addressing Low Recall

A significant limitation of the ASM based approach is the lower recall compared with other systems. Addressing low recall would be a necessary extension to any ASM based work. As discussed, one way is to use weakly labeled training data through distant supervision. Generally speaking, however, successful literature mining at the semantic level might require a combination of many approaches. An interesting area of future work is combining the shortest path kernel with other kernels such as the walk-based kernel[21] and the all-paths graph kernel[1].

References

[1] Airola, A., Pyysalo, S., Björne, J., Pahikkala, T., Ginter, F., and Salakoski, T. (2008). All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning. *BMC bioinformatics*, 9(Suppl 11):S2.

[2] Alpaydin, E. (2014). *Introduction to machine learning*. MIT press.

[3] Ananiadou, S., Kell, D. B., and Tsujii, J.-i. (2006). Text mining and its potential applications in systems biology. *Trends in biotechnology*, 24(12):571–579.

[4] Baumgartner, W. A., Cohen, K. B., Fox, L. M., Acquaah-Mensah, G., and Hunter, L. (2007). Manual curation is not sufficient for annotation of genomic databases. *Bioinformatics*, 23(13):i41–i48.

[5] Becker, K. G., Hosack, D. A., Dennis, G., Lempicki, R. A., Bright, T. J., Cheadle, C., and Engel, J. (2003). Pubmatrix: a tool for multiplex literature mining. *BMC bioinformatics*, 4(1):61.

[6] Björne, J. and Salakoski, T. (2011). Generalizing biomedical event extraction. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 183–191. Association for Computational Linguistics.

[7] Bunescu, R. and Mooney, R. (2007). Learning to extract relations from the web using minimal supervision. In *Annual meeting-association for Computational Linguistics*, volume 45, page 576.

[8] Bunescu, R. C. and Mooney, R. J. (2005). A shortest path dependency kernel for relation extraction. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 724–731. Association for Computational Linguistics.

[9] Chaussabel, D. and Sher, A. (2002). Mining microarray expression data by literature profiling. *Genome Biol*, 3(10):1–16.

[10] Choi, J. D. and McCallum, A. (2013). Transition-based dependency parsing with selectional branching. In *ACL (1)*, pages 1052–1062.

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

[12] Collins, M. and Duffy, N. (2001). Convolution kernels for natural language. In *Advances in neural information processing systems*, pages 625–632.

[13] Craven, M., Kumlien, J., et al. (1999). Constructing biological knowledge bases by extracting information from text sources. In *ISMB*, volume 1999, pages 77–86.

[14] De Marneffe, M.-C. and Manning, C. D. (2008). Stanford typed dependencies manual. *URL http://nlp. stanford. edu/software/dependencies manual. pdf.*

[15] GuoDong, Z., Jian, S., Jie, Z., and Min, Z. (2005). Exploring various knowledge in relation extraction. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 427–434. Association for Computational Linguistics.

[16] Hao, Y., Zhu, X., Huang, M., and Li, M. (2005). Discovering patterns to extract protein–protein interactions from the literature: Part ii. *Bioinformatics*, 21(15):3294–3300.

[17] Jenssen, T.-K., Lægreid, A., Komorowski, J., and Hovig, E. (2001). A literature network of human genes for high-throughput analysis of gene expression. *Nature genetics*, 28(1):21–28.

[18] Jiang, J. and Zhai, C. (2007). A systematic exploration of the feature space for relation extraction. In *HLT-NAACL*, pages 113–120.

[19] Kambhatla, N. (2004). Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, page 22. Association for Computational Linguistics.

[20] Kim, J.-D., Ohta, T., Tateisi, Y., and Tsujii, J. (2003). Genia corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(suppl 1):i180–i182.

[21] Kim, S., Yoon, J., and Yang, J. (2008). Kernel approaches for genic interaction extraction. *Bioinformatics*, 24(1):118–126.

[22] Krallinger, M. and Valencia, A. (2005). Text-mining and information-retrieval services for molecular biology. *Genome biology*, 6(7):224.

[23] Leroy, G. and Chen, H. (2005). Genescene: An ontology-enhanced integration of linguistic and co-occurrence based relations in biomedical texts. *Journal of the American Society for Information Science and Technology*, 56(5):457–468.

[24] Liu, H., Hunter, L., Kešelj, V., and Verspoor, K. (2013). Approximate subgraph matching-based literature mining for biomedical events and relations. *PloS one*, 8(4):e60954.

[25] MacKinlay, A., Martinez, D., Yepes, A. J., Liu, H., Wilbur, W. J., and Verspoor, K. (2013). Extracting biomedical events and modifications using subgraph matching with noisy training data. *ACL 2013*, page 35.

[26] McDonald, R., Pereira, F., Kulick, S., Winters, S., Jin, Y., and White, P. (2005). Simple algorithms for complex relation extraction with applications to biomedical ie. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 491–498. Association for Computational Linguistics.

[27] MEDLINE (2015). Key medline® indicators.

[28] Min, B., Grishman, R., Wan, L., Wang, C., and Gondek, D. (2013). Distant supervision for relation extraction with an incomplete knowledge base. In *HLT-NAACL*, pages 777–782.

[29] Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.

[30] Nebhi, K. (2013). A rule-based relation extraction system using dbpedia and syntactic parsing.

[31] Ono, T., Hishigaki, H., Tanigami, A., and Takagi, T. (2001). Automated extraction of information on protein–protein interactions from the biological literature. *Bioinformatics*, 17(2):155–161.

[32] Ravikumar, K., Liu, H., Cohn, J. D., Wall, M. E., Verspoor, K., et al. (2012). Literature mining of protein-residue associations with graph rules learned through distant supervision. *J. Biomedical Semantics*, 3(S-3):S2.

[33] Ring, H. Z., Kwok, P.-Y., and Cotton, R. (2006). Human variome project: an international collaboration to catalogue human genetic variation. *Pharmacogenomics*, 7(7):969–972.

[34] Silva, F. C. C. d., Valentin, M. D., Ferreira, F. d. O., Carraro, D. M., and Rossi, B. M. (2009). Mismatch repair genes in lynch syndrome: a review. *Sao Paulo Medical Journal*, 127(1):46–51.

[35] Stapley, B. J. and Benoit, G. (2000). Biobibliometrics: information retrieval and visualization from co-occurrences of gene names in medline abstracts. In *Pac Symp Biocomput*, volume 5, pages 529–540.

[36] Tanabe, L., Scherf, U., Smith, L., Lee, J., Hunter, L., and Weinstein, J. (1999). Medminer: an internet text-mining tool for biomedical information, with application to gene expression profiling. *Biotechniques*, 27(6):1210–4.

[37] Tomanek, K., Wermter, J., and Hahn, U. (2007). Sentence and token splitting based on conditional random fields. In *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics*, pages 49–57.

[38] van der Eijk, C. C., van Mulligen, E. M., Kors, J. A., Mons, B., and van den Berg, J. (2004). Constructing an associative concept space for literature-based discovery. *Journal of the American Society for Information Science and Technology*, 55(5):436–444.

[39] Verspoor, K., Cohen, K. B., Lanfranchi, A., Warner, C., Johnson, H. L., Roeder, C., Choi, J. D., Funk, C., Malenkiy, Y., Eckert, M., et al. (2012). A corpus of full-text journal articles is a robust evaluation tool for revealing differences in performance of biomedical natural language processing tools. *BMC bioinformatics*, 13(1):207.

[40] Verspoor, K., Yepes, A. J., Cavedon, L., McIntosh, T., Herten-Crabb, A., Thomas, Z., and Plazzer, J.-P. (2013). Annotating the biomedical literature for the human variome. *Database*, 2013:bat019.

[41] Wikipedia (2014). Biocurator — wikipedia, the free encyclopedia. [Online; accessed 31-May-2015].

[42] Wilbur, J., Smith, L., and Tanabe, L. (2007). Biocreative 2. gene mention task. In *Proceedings of Second BioCreative Challenge Evaluation Workshop*, pages 7–16.

[43] Witten, I. H. (2005). Text mining. *Practical handbook of Internet computing*, pages 14–1.

[44] Wren, J. D., Bekeredjian, R., Stewart, J. A., Shohet, R. V., and Garner, H. R. (2004). Knowledge discovery by automated identification and ranking of implicit relationships. *Bioinformatics*, 20(3):389–398.

[45] Zelenko, D., Aone, C., and Richardella, A. (2003). Kernel methods for relation extraction. *The Journal of Machine Learning Research*, 3:1083–1106.