Playing hide and seek on the genomic playground

Unveiling biological function from literature

Sofie Van Landeghem

Promotor: Prof. Dr. Y. Van de Peer
Co-promotor: Prof. Dr. B. De Baets
Co-promotor: Dr. Y. Saeys

Ghent University
Faculty of Sciences
Department of Plant Biotechnology and Bioinformatics
VIB Department of Plant Systems Biology
Bioinformatics and Systems Biology

This research was made possible by funding from the FWO Vlaanderen (Fonds voor Wetenschappelijk Onderzoek in Vlaanderen) and BOF (Bijzonder Onderzoeksfonds).

Dissertation submitted in fulfillment of the requirements for the degree of Doctor (PhD) in Sciences, Bioinformatics.

Academic year 2011-2012

Examination committee

Prof. Dr. Geert De Jaeger (Chairman)
Dept. of Plant Biotechnology and Bioinformatics, Faculty of Sciences, Ghent University

Prof. Dr. Yves Van de Peer (Promotor)
Dept. of Plant Biotechnology and Bioinformatics, Faculty of Sciences, Ghent University

Prof. Dr. Bernard De Baets (Co-promotor)
Dept. of Mathematical Modelling, Statistics and Bio-informatics,
Faculty of Bioscience Engineering, Ghent University

Dr. Yvan Saeys (Co-promotor, reading commission)
Dept. of Plant Biotechnology and Bioinformatics, Faculty of Sciences, Ghent University

Dr. Pierre Rouzé (Reading commission)
Dept. of Plant Biotechnology and Bioinformatics, Faculty of Sciences, Ghent University

Prof. Dr. Peter Dawyndt (Reading commission)
Dept. of Applied Mathematics and Computer Science, Faculty of Sciences, Ghent University

Prof. Dr. Veronique Hoste (Reading commission)
Faculty of Translation Studies, University College Ghent

Dr. Ir. Tim Van den Bulcke (Reading commission)
Antwerp University Hospital

Dr. Filip Ginter
Dept. of Information Technology, University of Turku (Finland)

Acknowledgments - Dankwoord

Writing this acknowledgment section is probably the easiest task of all, as there are so many wonderful people in my life I could easily fill two chapters.

First of all, I want to thank Yvan Saeys, Chris Cornelis and Rafi Benotmane, who were in charge of supervising my Master's thesis 5 years ago, and whose excellent guidance has inspired me greatly. They supported my enthousiasm by allowing me to pursue the direction of the research that interested me most, which would ultimately be the foundation of the text mining studies presented in this PhD thesis.

Next, I want to express my enourmous gratitude to Yves Van de Peer, for offering me the opportunity to be a PhD student within his research group, for giving me the freedom and confidence to work on my beloved text mining, and for simply being a wonderful boss with a great sense of humor. I have thoroughly enjoyed working within our bioinformatics group, and hope my time here is not quite over yet...

Additionally I want to thank Bernard De Baets and Yvan Saeys for co-supervising my PhD research, and Geert De Jaeger, Pierre Rouzé, Peter Dawyndt, Veronique Hoste, Tim Van den Bulcke and Filip Ginter to agree to be a member of the examn commission and for carefully reading (and judging) my thesis. I have highly appreciated your feedback and hope I've improved the thesis accordingly.

Thanks to Thomas Abeel for extremely efficient and pleasant collaborations, to Thomas Van Parys, Kenny Billiau, Michiel Van Bel, Marijn Vandevoorde and Frederik Delaere for technical support, and to all other *binaries* for releasing the inner nerd (of whom, I shall not specify) on many hilarious occassions (*work, work*). Ofcourse many thanks also to the other members of the group and department for many fun parties and lunch dates.

Crossing the baltic sea, *paljon kiitoksia* to Tapio Salakoski for a warm welcome in Turku, and to Filip Ginter and Laura Halme for making me feel at home in the cold Finnish winter wonderland. I think EVEX was worth all the deadline rushes and occasional desperation! Also thanks to everyone I've collaborated with, especially Jari Björne, Sampo Pyysalo and Tomoko Ohta, and to the BioNLP community as a whole for being such a wonderful group of people who are truely interested in advancing the science rather than be consumed by competition and impact factors. Attending the ACL conferences and BioNLP workshops has always been a very enjoyable experience.

Er is verder nog een rits vrienden die hier een vermelding verdienen voor de vele fijne, en broodnodige, ontspannende momenten. Bedankt aan m'n ex-studiegenootjes Anne, Vicky, Sofie en Davy, de vrienden van de badminton en van OCC, en tenslotte de vrienden uit het waasland en Antwerpen. Ik durf hier geen uitgebreide opsomming van namen te riskeren, maar jullie weten vast wie jullie zijn en dat ik jullie graag zie :-)

Bedankt ook aan mijn geweldige zussen, lieve schoonbroers, en aan Raf en Ria voor de morele steun. Een speciale *merci* voor mijn ouders, die me van van kinds af aan alle kansen hebben gegeven in het leven die me op dit punt brachten.

Tenslotte rest me nog Stephen te bedanken en dit doctoraat aan hem op te dragen. Zonder jou was ik nooit de persoon geworden die ik vandaag ben en ik ben je eeuwig dankbaar voor je nooit aflatende, onvoorwaardelijke steun en liefde.

Gent, April 2012
Sofie Van Landeghem

Table of contents

Samenvatting
- Dutch summary-

Onderzoek naar specifieke *natural language processing* technieken voor biomoleculaire tekst (BioNLP) is ontstaan uit de noodzaak aan automatische methoden om de grote hoeveelheid bestaande literatuur efficiënt te analyseren. Grootschalige *tekst-mining* projecten kunnen biomedisch onderzoek immers ondersteunen in tal van applicaties, zoals het aanvullen van bestaande databanken, het koppelen van nieuwe experimentele resultaten aan bestaande kennis en het genereren van nieuwe onderzoekshypothesen. In deze thesis bespreken we voornamelijk de automatische extractie van gekende associaties tussen genen en proteïnen om dergelijke applicaties mogelijk te maken.

Proteïne-proteïne interacties

De inherente complexiteit van natuurlijke taal belemmert een accurate extractie van tekstuele informatie. Eén van de eerste en best bestudeerde uitdagingen in het BioNLP domein is de automatische extractie van proteïne-proteïne interacties (PPIs). Een gedetailleerd literatuuronderzoek over dit onderwerp bracht aan het licht dat er geen ingeburgerde standaarden bestaan voor de evaluatie van PPI extractiemethoden. We brachten een aantal fundamentele keuzemogelijkheden bij dergelijke evaluaties in kaart, evenals hun invloed op de gerapporteerde performantie van de bestudeerde systemen. Bovendien werd een aantal praktische regels opgesteld die het uitvoeren van vergelijkende studies zouden moeten vereenvoudigen.

Vervolgens ontwikkelden we een nieuw *machine learning* systeem voor de tekstuele extractie van PPIs. Dit systeem analyseert zowel de lexicale als de syntactische en grammaticale informatie in een zin en maakt hiervan een synthese in rijke *feature* vectoren. Verder voerden we de eerste analyse van *feature selectie* methoden in dit domein uit, resulterend in meer efficiënte classificatiemodellen. Tenslotte werd de nieuwe PPI extractiemethode geëvalueerd via cross-dataset experimenten, die een meer realistische kijk op de tekst-mining performantie bieden.

Extractie van events

Het extraheren van ongerichte binaire relaties, zoals PPIs, biedt onvoldoende potentieel om complexe biomoleculaire interacties tussen genen en proteïnen te bevatten. Bijgevolg is men binnen het BioNLP domein overgeschakeld naar een expressievere *event* representatie, onder invloed van de *BioNLP Shared Task on Event Extraction*. Events hebben een specifiek type en polariteit, hun argumenten nemen verschillende semantische rollen aan en zij treden op in een speculatieve, dan wel affirmatieve context. Eventtypes omvatten zowel fysieke interacties zoals fosforylatie en genexpressie, als recursief gedefinieerde regulaties.

Voor deze nieuwe uitdaging hebben we het eerder geïntroduceerde systeem voor PPI extractie substantieel uitgebreid. Hierbij werd vooral getracht om verkeerde voorspellingen te voorkomen en zo een hoge accuraatheid van het systeem te garanderen. Verschillende technieken werden getest, zoals allerlei SVM *kernels*, feature selectie en methoden voor het verwerken van de invoer- en uitvoerdata. Voor de detectie van speculatie en negatie werd een regelgebaseerd systeem ontworpen. Van de 24 deelnemende groepen in de BioNLP Shared Task van 2009 behaalde ons systeem een vijfde plaats met 33.41% recall, 51.55% precisie en 40.54% F-score. In een vervolgstudie werd het systeem nog verder geoptimaliseerd, wat resulteerde in een relatieve performantiestijging van 10%. De uiteindelijke resultaten bedragen aldus 37.43% recall, 54.81% precisie en 44.48% F-score.

De complexiteit en onvoorspelbaarheid van machine learning algoritmen verhindert vaak een intuïtief inzicht in de gemaakte predicties. Dankzij feature selectie kunnen we echter de belangrijkste features identificeren en op die manier de eigenschappen van de classificatiemethoden beter doorgronden. Bijgevolg kunnen betere methoden ontwikkeld worden en worden de resultaten van het systeem begrijpelijker voor de eindgebruiker. De recent ontwikkelde methode van *ensemble feature selectie* werd aldus toegepast op event extractie, waarbij een beter inzicht werd bekomen in de werking van de algoritmen. We bespreken verschillende voorbeelden van belangrijke biologische en taalkundige features, evenals de conclusies bekomen uit deze analyses.

Relaties tussen entiteiten

Een bijkomende taak van de BioNLP Shared Task omvat de extractie van non-causale of *entiteit* relaties. Dergelijke relaties identificeren bijvoorbeeld genpromotors en proteïnecomplexen, zodat een nauwkeuriger voorstelling van tekstuele beschrijvingen mogelijk gemaakt wordt. Om dergelijke entiteitrelaties te extraheren, werden semantische algoritmen gecombineerd met machine learning en feature selectie. Dit systeem behaalde een tweede plaats in de BioNLP Shared Task van 2011, met 37.04% precisie, 47.48% recall en 41.62% F-score.

Bovendien werd dit systeem vergeleken met het winnende systeem van de universiteit van Turku (57.7% F-score). Het performantieverschil tussen de twee systemen

werd geanaliseerd op een meer extensieve en gerelateerde dataset. Bovendien werd geëxperimenteerd met de intersectie en de unie van de predicties, zodat respectievelijk hoge precisie en hoge recall bekomen wordt. Tenslotte bespreken we een specifiek deelresultaat waarbij zeer hoge performantie gemeten werd (F-score boven 90%) en dat essentieel is voor de integratie van tekst-mining met databanken.

Tenslotte worden entiteitrelaties toegepast voor het verbeteren van event extractie. De resultaten van deze analyse onderstrepen de noodzaak om de specifieke eigenschappen van de verschillende eventtypes in acht te nemen bij het ontwikkelen van automatische extractie methoden.

EVEX: een grootschalige tekst-mining databank

Om tekstuele data te integreren met bestaande databanken, is het noodzakelijk dat tekstmining wordt toegepast op grote schaal en dat de resultaten gelinkt worden aan informatie uit databanken zoals die van NCBI, UniProt, KEGG en BioGRID.

We presenteren de eerste grootschalige studie die de automatische extractie van events combineert met een algoritme voor gennormalisatie, dat ambigue gensymbolen linkt aan unieke databank IDs. Deze *pipeline* bestaat uit *state-of-the-art* componenten die geëvalueerd werden op internationale competities. Ze werd toegepast op alle 21 miljoen beschikbare PubMed abstracts en de 372 duizend vrij verwerkbare artikels uit PubMed Central. De resulterende dataset -EVEX genoemd- bevat meer dan 34 miljoen moleculaire events tussen 67 miljoen gensymbolen die gelinkt kunnen worden aan meer dan 120 duizend genen van 4800 organismen. De dataset bevat informatie over virussen, bacteriën, schimmels, planten en dieren.

De EVEX dataset werd verder uitgebreid met genfamilies en abstracte generalisaties, rekening houdend met lexicale varianten en synoniemen. Deze integratie opent interessante mogelijkheden voor het genereren van nieuwe hypothesen gebaseerd op homologie. Alle geëxtraheerde events en hun generalisaties zijn publiek beschikbaar als een MySQL databank.

Bovendien werd een intuïtieve webapplicatie ontwikkeld die manuele verkenning van tekst-mining resultaten mogelijk maakt zonder à priori BioNLP kennis te vereisen. Deze webapplicatie vat de gekende informatie over een gegeven gen samen en presenteert indirecte associaties tussen twee genen zoals co-regulatie.

Concrete toepassingen

Tenslotte bespreken we het nut van de eventgebaseerde tekst-mining aanpak voor toepassingen als databank-ondersteuning en (re)constructie van moleculaire netwerken. Deze mogelijkheden worden geïllustreerd aan de hand van een studie over het metabolisme van de bacterie *E. coli*. Deze analyses belichten een aantal interessante toekomstige opportuniteiten voor de integratie van tekst-mining met bestaande moleculaire data.

Summary

The field of natural language processing for biomolecular texts (BioNLP) aims at large-scale text mining in support of life science research. Its primary motivation is the enormous amount of available scientific literature, which makes it essentially impossible to rapidly gain an overview of prior research results other than in a very narrow domain of interest. Among the typical use cases for BioNLP applications are support for database curation, linking experimental data with relevant literature, and hypothesis generation. This thesis discusses the extraction of information about known associations between genes and proteins to support such use cases.

Protein-protein interactions

Due to the intrinsic complexity of natural language, accurately extracting information from text is a challenging discipline. As one of the first problems addressed by the BioNLP community, the extraction of protein-protein interactions (PPIs) has been widely studied and many different predictive frameworks proposed. During literature review of these methods, it has become clear that this field is still struggling with a heterogeneous collection of datasets, data formats and evaluation methods. Several fundamental evaluation problems are discussed, including their influence on the reported performance rates. A set of practical guidelines is also proposed to ensure a meaningful evaluation.

Further, a novel machine learning framework was developed for PPI extraction from text. This framework analyses both the lexical and syntactic information from sentences and synthesizes all this information in rich feature vectors. We present the first extensive analysis of applying fully automated feature selection in this domain, obtaining more cost-effective models. Finally, our PPI extraction technique was evaluated on several novel cross-dataset experiments, offering a more realistic view on model performance.

Event extraction

Recognizing that extraction of undirected binary relations such as PPIs do not provide sufficient detail for representing complex biomolecular interactions, the focus has shifted towards a more detailed analysis of the textual statements. This approach was formalized as an *event extraction* task and greatly popularized in the series of BioNLP Shared Tasks

on Event Extraction. The detection of biomolecular events from text includes various physical events such as phosphorylation and gene expression, as well as recursively defined regulatory events. Their extraction includes additional vital information such as the type and polarity of the relationship, the identification of the semantic roles of the participating entities and whether it was stated in a speculative or affirmative context.

A detailed account of the extension of our machine-learning framework is presented, employing a set of type-specific classifiers run in parallel for event extraction. Our work is mainly focused around the filtering of false positives, creating a high-precision extraction method. Various different techniques were tested such as different SVM kernels, feature selection and filters for data pre- and post-processing. To detect negation and speculation in text, a rule-based system was implemented; simple in design, but effective in performance. Our framework ranks 5^{th} out of 24 international teams in the BioNLP Shared Task of 2009, achieving 33.41% recall, 51.55% precision and 40.54% F-score. Follow-up studies further improved the method and a relative performance gain of 10% was obtained, resulting in 37.43% recall, 54.81% precision and 44.48% F-score.

Black-box behavior of machine learning systems currently limits understanding of the true nature of the predictions. However, feature selection is capable of identifying the most relevant features in any supervised learning setting, providing insight into the specific properties of the classification algorithm. This allows building more accurate classifiers while at the same time bridging the gap between the black-box behavior and the end-user who has to interpret the results. The novel method of ensemble feature selection is applied to the event extraction challenge, discarding a large fraction of machine generated features and improving classification performance. Furthermore, we present numerous examples of highly discriminative features that model either biological reality or common linguistic constructs, illustrating how feature selection can be used to gain understanding in automatically generated predictions. Finally, we discuss a number of insights from these analyses that may help improving current text mining tools.

Entity relations

One of the supporting tasks of the BioNLP Shared Task, designed to provide more fine-grained text predictions, is the extraction of non-causal or 'entity' relations. Such entity relations between genes and domain terms identify the relations between genes, promoters, complexes and various other molecular entities found in text, enabling an enhanced representation of the biological processes underlying textual statements. We have implemented an extraction system for such non-causal relations between genes and domain terms, applying semantic spaces, machine learning and feature selection techniques. Our system ranks second in the official results of the BioNLP Shared Task of 2011, achieving 37.04% precision, 47.48% recall and 41.62% F-score.

Further, our framework is compared with the system ranking first, developed by the University of Turku (57.7% F-score). We investigate the performance discrepancy by

analysing the influence of predicted domain terms, using a related and more extensive dataset. Additionally, a hybrid system is constructed, combining the two frameworks and experimenting with intersection and union combinations for respectively high-precision and high-recall predictions. Finally, extremely high-performance results (F-score above 90%) are highlighted, representing a specific subclass of embedded entity relations that are essential for integration of text mining predictions with database facts.

Finally, we present the first study of applying entity relations for enhancing event extraction performance. While obtaining promising results, we argue that an event extraction framework benefits most from this new data when taking intrinsic differences between various event types into account.

EVEX: a large-scale text mining resource

To enable full integration of textual data with existing biomolecular databases, it is crucial that text mining tools scale up to millions of articles and their results can be unambiguously linked to data records from authoritative resources such as NCBI, UniProt, KEGG and BioGRID.

We present the first bibliome-wide study that combines automated extraction of complex biomolecular events with a gene normalization system that maps ambiguous gene mentions in text to unique gene identifiers. This pipeline, consisting of state-of-the-art components that were thoroughly evaluated on two highly relevant community-wide challenges, was applied to all 21 million PubMed abstracts and all 372 thousand PubMed Central open-access full-text articles. The resulting dataset, called EVEX, contains more than 34 million biomolecular events among 67 million gene mentions that could be linked to more than 120 thousand distinct genes from over 4800 species covering the full taxonomic tree, including viruses, bacteria, fungi, plants and animals.

The data was further enriched with gene family data, providing interesting opportunities for homology-based hypothesis generation. Further, abstract generalizations accounting for lexical variants and synonymy. The originally extracted event occurrences, as well as their generalized variants, are publicly available as a MySQL database.

Further, an intuitive web application is developed, allowing explorative browsing of the EVEX text mining results without prior knowledge on BioNLP. This web application allows for knowledge summarization on any given gene as well as retrieval of indirect associations between two genes, such as co-regulation.

Real world applications

Finally, we discuss the applicability of event-based text mining tools for database and pathway curation. These opportunities are illustrated on a specific use case involving NADP(H) metabolism in *E. coli*. The analyses show promising results and highlight interesting future prospects.

1 Introduction

The discovery in 1953 of the double helix structure of DNA was followed by valuable insights into the genetic code only a decade later. These findings ignited the molecular revolution, establishing **molecular biology** as a new and exciting research field that, to this day, uncovers new secrets concerning the mechanisms of life.

At the junction of molecular biology and computer science arises the discipline of **bioinformatics**, offering algorithms and tools to process the ever increasing amount of biological data generated in experimental studies. Its subdomains are numerous and include genome annotation, systems biology and comparative genomics. This thesis specifically focuses on data mining and text mining tools for the life sciences.

1.1 Molecular biology

Understanding the mechanisms of life requires a thorough analysis of the structure and function of individual genes and proteins, as well as their physical interactions and regulatory processes. In this section we provide some basic insights into these mechanisms, as they form the extraction target of the text mining work described in this thesis.

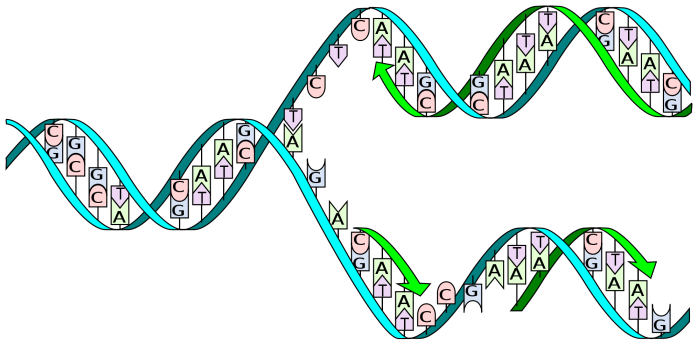


Figure 1.1: DNA replication: the two strands of the DNA helix are broken up and used as templates for new copies. Picture from Wikimedia Commons.

1.1.1 From DNA and genes to RNA and proteins

DNA (deoxyribonucleic acid) is the most important carrier of genetic information, and encodes this information through specific sequences of four different nitrogen-based molecules (nucleobases): adenine (A), cytosine (C), guanine (G) and thymine (T). The DNA molecule is composed of two long strands (polymers) of nucleotides; a combination of a nucleobase, a sugar and a phosphate group. The two polymers are structured as a double helix, connected to each other through hydrogen bonds. These hydrogen bonds link two complementary nucleobases together, with A bonding only to T, and C only to G. The resulting A-T and C-G base pairs effectively duplicate the genetic information across the two anti-parallel strands of the DNA helix. Their sequence determines the DNA sequence, which encodes all hereditary information (the genome) in almost all organisms, with the exception of some viruses that employ RNA (ribonucleic acid) as genetic material.

During cell division, an organism's DNA is replicated by first separating the two polymers of the DNA strand by breaking the hydrogen bonds between the nucleobases (Figure 1.1). Subsequently, free nucleotides within the cell bind to the template strands, successfully creating two copies of the DNA sequence. DNA replication is the basis for biological inheritance as it enables transferring the entire genome to a daughter cell of an organism and, consequently, to its offspring. The DNA of an organism is typically structured into chromosomes. Humans, for example, have 23 pairs of chromosomes and within each pair, one is inherited from the mother and one from the father.

The individual portions of a DNA sequence that code for a specific function are called **genes**, the basic units of hereditary. These genes are transcribed into RNA. **Func-**

tional RNA or 'non-protein-coding' RNA plays an important role in various cellular processes. RNA molecules that are further translated to amino acids using the genetic code, give rise to proteins. **Proteins** are essential molecules of any organism, performing a wide variety of functions: from catalysis of biochemical reactions to structural support, cell signaling and immune response.

1.1.2 Gene expression and regulation

While the full genome is present in each cell of the body, the production or synthesis of RNA and proteins from genes depends on environmental conditions. **Gene regulation** determines how and when genes are expressed, giving rise to proteins and functional RNA. For example, so-called housekeeping genes are essential for basic cellular functions and are always expressed, while others may only be expressed in a particular cell-type (e.g. muscle cells), or through the activation of particular signals (e.g. hormones).

Gene expression is tightly regulated through various mechanisms at different stages. **Epigenetic changes** are chemical modifications to DNA and to histone proteins, which are essential for packaging the DNA into chromosomes (Bernstein *et al.*, 2007). Such epigenetic changes often arise during development and form an additional layer of information on top of the DNA sequence. These cell-type specific changes may be preserved during cell division and may even last for multiple generations. They can directly influence gene expression levels by modifying the accessibility of the DNA to proteins that would bind to it, such as transcription factors.

Transcription denotes the process of creating an RNA copy of the DNA sequence. By breaking the hydrogen bonds of the base pairs between the two DNA strands, a complementary messenger RNA (mRNA) copy of the gene is built, with uracil (U) fulfilling the role of thymine as the preferred binding partner of adenine. Transcription factors can either activate or repress gene expression by binding to a specific DNA sequence. Often genes have several such binding sites for distinct transcription factors, requiring the co-operation of several different transcription factors for their expression.

Post-transcriptional modifications further convert the pre-mRNA transcript to mature mRNA through RNA-splicing. RNA-splicing removes the nucleotides of the mRNA sequence (introns) that are not translated to amino acids, keeping only the exons ('expressed regions'). However, occasionally these exons can be combined in various ways, resulting in different mRNA products from a single gene (alternative splicing).

Translation then determines the unique amino acid sequence from the mature mRNA molecule through the genetic code, which maps three nucleotides (a codon) to one of 20 possible amino acids (Figure 1.2).

The final linear chain of amino acids determines the protein. These proteins are

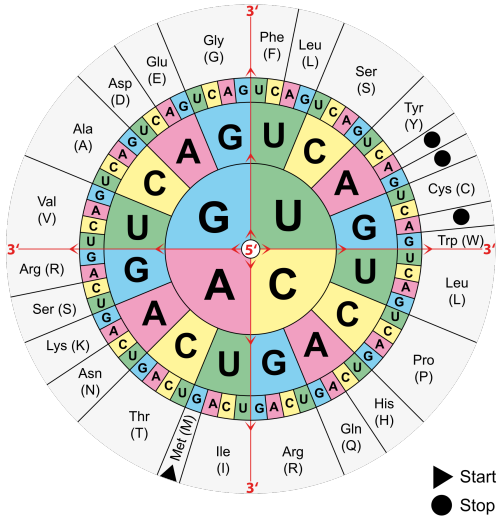


Figure 1.2: The genetic code that translates a codon into an amino acid. Picture from Wikimedia Commons.

folded into three-dimensional structures and targeted to the appropriate location within the cell to perform their desired function. **Post-translational modifications** (PTMs) further control the function of these proteins by chemical modifications. Phosphorylation is one of the most common PTM mechanisms and allows for activation or deactivation of the protein under certain circumstances. We refer to Baginsky *et al.* (2010) for more background information on gene expression and regulation.

1.1.3 Physical and functional interactions

To understand the complex machinery in living cells, systems biology approaches often visualize regulatory links and physical interactions as interaction networks. Depending on the specific interaction type, various large-scale networks may be constructed.

Protein-protein interactions give rise to 'molecular machines' (complexes) consisting of a large number of interacting protein components, often necessary to undertake certain biological functions. The nature of this function determines whether the physical contact between the proteins is static or permanent.

Other types of physical interactions are **protein-DNA** or protein-gene interactions. For example, transcription factors are proteins with specific DNA-binding domains, ac-

tivating or inhibiting the transcription of genes by binding to enhancer or promoter regions of DNA close to the regulated genes. Data on transcription factor bindings is often combined with expression data to analyse co-regulation and functional modules.

Phosphorylation is a **chemical modification** of a protein by a protein kinase, a type of enzyme that adds a phosphate group to the target protein. Such phosphate groups may again be removed by a phosphatase, a different type of enzyme. Large-scale phosphorylation networks may help elucidating signaling pathways.

Finally, a more indirect interaction between genes occurs when two genes affect each other's function. For instance, the mutation of a single gene may be viable, reflecting the robustness of a biological system, but in combination with another mutation, becomes lethal. Such functional relationships are expressed through **genetic interactions**, and are crucial for a thorough understanding of biological pathways.

We refer to Zhu *et al.* (2007) for a more in depth overview of different biomolecular interaction types and their corresponding networks.

1.2 Bioinformatics

As increasingly more research interest turned to molecular biology in the second half of the previous century, the data and findings of experimental studies started to pile up. Such studies would for example aim at elucidating the relations between certain genotypes and a resulting phenotype, i.e. a set of observable traits of a particular organism. During the last few decades, advances in biotechnology further led to large-scale sequencing projects and immense datasets on gene expression patterns and molecular interactions. Bioinformatics tools and resources have thus become a true necessity to process the abundance of data in the life sciences.

1.2.1 Comparative genomics

Experimental studies are often conducted on model organisms, usually chosen because they have a short life-cycle or are easy to manipulate experimentally. Such model organisms include *Escherichia coli* (a bacterium), *Arabidopsis thaliana* (a flowering plant), *Saccharomyces cerevisiae* (yeast), *Drosophila melanogaster* (fruit fly) and *Mus musculus* (mouse). Valuable information determined by studying these species can be transferred to other organisms that are more difficult to study directly. For example, genes with high sequence similarity (homologs) originate from a common ancestor gene, either through a speciation event (orthologs) or through a duplication event (paralogs). Orthologs in particular often perform the same function in different organisms.

1.2.2 Data resources

To structure and easily query available experimental data, various bioinformatics resources were designed. Some of these target a specific model organism such as MGI for mouse (Blake *et al.*, 2011) and TAIR for *Arabidopsis* (Lamesch *et al.*, 2012), while others are mainly concerned with a distinct biomolecular event type such as BioGRID for protein-protein interactions (Stark *et al.*, 2011) and P3DB for phosphorylation (Gao *et al.*, 2008).

Most data resources consist of a mixture of both primary data and meta-data. Primary genetic data refers to the original, raw data such as sequence data and gene expression data. Meta-data on the other hand are generated from these raw data sources using gene models and other predictive methods. While meta-data is much more abundantly present in the various data resources, it is important to realise that this data may contain errors. A data validation step is thus necessary when incorperating such data in large-scale projects.

In this thesis, we refer mainly to databases (DBs) hosted at the NCBI, a rich resource for all sorts of data concerning genes and proteins, DNA and chemicals, homology, taxonomy classification and diseases (Sayers *et al.*, 2010). The Taxonomy DB contains curated classification and nomenclature records for all organisms in public sequence databases, representing ca. 10% of the described species of life on the planet. Entrez Gene is the *de facto* cross-species gene nomenclature authority, containing more than 8 million Entrez Gene identifiers (EG IDs) from over 8,000 different taxa. Finally, nucleotide sequences and their protein translations are available from several sources, including GenBank and RefSeq.

Another important resource containing protein sequences and functional information is UniProt, providing more than 530 thousand manually curated sequences and 18.5 million automatically annotated sequence entries (The UniProt Consortium, 2011). Further, Ensembl is a huge resource for valuable information on vertebrate genomic data (Flicek *et al.*, 2011) and provides additional support for metazoa, plants, protists, fungi, and bacteria (Kersey *et al.*, 2010). Gene Ontology (GO) is a widely used resource that links genes to terms from structured vocabularies, covering cellular components, biological processes and molecular functions (The Gene Ontology Consortium, 2008). GO annotations are often used to identify groups of similar genes within a certain dataset. GO is part of the bigger project Open Biomedical Ontologies (OBO), which aims as maintaining controlled vocabularies in the biomedical domain.

Finally, two important large-scale literature resources are available through the NCBI website: PubMed and PubMed Central. PubMed (PM) provides access to more than 21

million citations of biomedical literature[1], and is still growing exponentially. PubMed Central (PMC) provides 2.3 million full-text articles, of which 400 thousand are within the Open Access (OA) subset and thus available for automated text mining algorithms[2].

1.2.3 Data integration

The volume of the datasets discussed in the previous section illustrate the need for data and text mining techniques. Data integration of different resources is crucial for a full overview on all available knowledge on a certain gene or biological process. To enable meaningful data integration, it is necessary to unambiguously link entities from one database to another, recognizing equality of e.g. a UniProt protein sequence and a GenBank sequence. Mapping schemes exist that identify the relations between authorative resources such as Entrez Gene, UniProt and Ensembl. These identifiers are widely used by other resources to unambiguously link the existing data together.

However, to integrate text mining results with database facts, it is necessary to identify a gene identifier for a given textual gene symbol. This non-trivial challenge is further discussed in Section 1.3.4.

1.3 BioNLP

The field of natural language processing for biomolecular texts (BioNLP) aims at large-scale text mining in support of life sciences research. Its primary motivation is the enormous amount of available scientific literature, which makes it essentially impossible to rapidly gain an overview of prior research results other than in a very narrow domain of interest (Krallinger and Valencia, 2005).

To analyse textual data and produce formal representations of a certain extraction target (Section 1.3.1), several steps are necessary. First, a set of documents relevant to the challenge needs to be collected (Section 1.3.2). Next, is it is necessary to recognise (Section 1.3.3) and unambiguously identify (Section 1.3.4) the concepts or entities in text that are of interest for the specific text mining goal. Subsequently, the text is transformed to a more formal representation by applying both lexical (Section 1.3.5) as well as syntactic (Section 1.3.6) analyses. The main focus of this thesis concerns the final step of relation extraction itself (Section 1.3.7).

[1] `http://www.ncbi.nlm.nih.gov/pubmed/`
[2] `http://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/`

1.3.1 Extraction target

One of the first challenges addressed by research in the BioNLP field was the extraction of protein-protein interactions (PPIs) from text. This extraction challenge is highly relevant as proteins often form larger molecular machines to perform certain biological functions. Additionally, other relations such as genetic interactions and chemical modifications are sometimes also included (Section 1.1.3). Chapter 3 further discusses the extraction of PPIs from text.

The recent introduction of event extraction has further broadened the scope of BioNLP extraction targets. Events are more expressive representations of biomolecular interactions and they cover a wider range of biological processes, including protein metabolism (e.g. transcription and catabolism), protein modification (e.g. phosphorylation) and fundamental molecular events (e.g. binding and localization). On top of these physical events, causal relations are represented as regulatory events with a certain polarity (positive, negative or neutral). Chapter 4 details the extraction of biomolecular events from text.

Both PPI and event extraction are concerned with relations between genes and proteins. In contrast, an additional task of extracting entity relations has recently been introduced, defining non-causal relationships between genes/proteins and general domain terms. They include subunit-of relations (e.g. a protein complex) and part-of relations (e.g. a specific DNA site on a gene). Chapter 5 discusses the extraction of entity relations and their applicability for event extraction.

While there are many more possible extraction challenges in the field of BioNLP, such as gene-disease relations or phenotypic changes, these topics are not covered in this thesis. However, the methods presented in this work are sufficiently general to be applicable to any new, well-defined extraction target.

1.3.2 Information retrieval

By information retrieval (IR), we denote the first step in the text mining pipeline that concerns the initial data collection. This step heavily depends on the specific extraction target and use case. Outside the domain of BioNLP, search engines such as Google[3] or Bing[4] are widely used to retrieve relevant documents on the world wide web. Similarly, the NCBI databases are all accessible through a unified search portal called Entrez[5], while the Gene Ontology can be browsed by means of AmiGO[6].

[3] http://www.google.com
[4] http://www.bing.com/
[5] http://www.ncbi.nlm.nih.gov/Entrez/
[6] http://amigo.geneontology.org/

While search engines provide fast access to information through manual browsing, they are insufficient for large-scale data mining or data integration projects. For this purpose, application programming interfaces (APIs) are usually designed for automated querying of the database. NCBI for example has designed the Entrez Programming Utilities (eUtils), offering access to all NCBI databases through retrieval scripts. Finally, many large-scale resources provide a publicly accessible FTP server for bulk download.

For automated text mining analyses, documents should typically consist of raw text. When only PDF or HTML formats are available, these files are first processed and transformed into a machine-readable format such as XML.

1.3.3 Entity recognition

Once the relevant documents are retrieved, a text mining system needs to search for the molecular objects (entities) of interest. Often, these entities are assigned a specific name such as a gene symbol. In this thesis, the named entity recognition (NER) step refers mainly to the retrieval of genes and gene products (GGPs), which are the main participants in both the PPI and the event extraction challenge.

To extract entity relations, it is further necessary to identify non-named entities. Such entities are specific domain terms referred to by general English words such as 'promoter' or 'complex'.

1.3.4 Named entity normalization

Named entity normalization (NEN) is the procedure by which named entities are assigned a unique identifier. Due to the lack of community-wide approved standards for assigning gene symbols (Chen *et al.*, 2005), this is not a trivial task. Authors often introduce their own lexical variants or abbreviations for specific genes.

Synonymy results in a plethora of different textual mentions that map to the same gene ID. For example, *Esr-1*, *ER alpha* and *NR3A1* all refer to the *Estrogen receptor 1* gene. On the other hand, abbreviations result in different possible identifiers for one specific mention. For example, both the *Estrogen receptor* gene as well as the *Enhancer of shoot regeneration* gene are occasionally abbreviated to *esr*. Finally, gene nomenclature exhibits high inter-species ambiguity, as orthologs are often assigned to the same name. For example, the *Esr-1* gene has known orthologs in human, mouse, rat and zebra fish.

As a central challenge in biological text mining, gene normalization has been a major focus of BioCreative, the longest-running community-wide challenge in the domain (Hirschman *et al.*, 2005; Krallinger *et al.*, 2008; Leitner *et al.*, 2010; Arighi *et al.*, 2011).

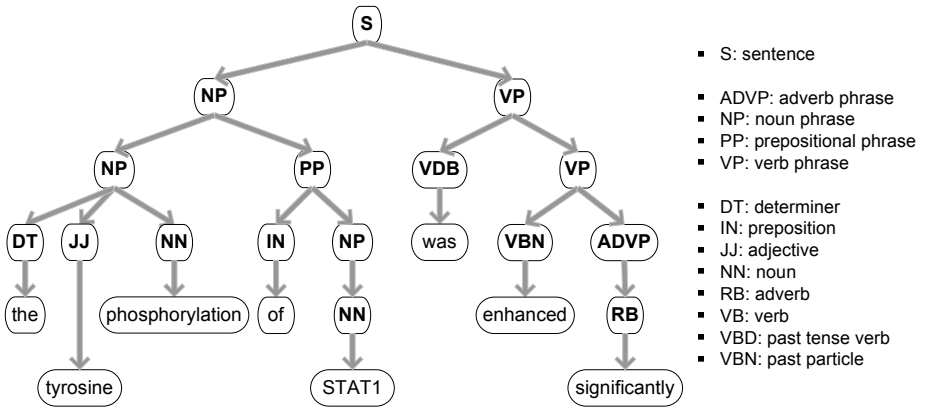


Figure 1.3: Constituency tree for the sentence 'The tyrosine phosphorylation of STAT1 was enhanced significantly'. The tree divides the sentence into large (noun/verb) phrases and breaks it down further until each word (leaf) gets a part-of-speech tag (direct parent of the leaf). Frequently occurring abbreviations are listed on the right (Santorini, 1990).

1.3.5 Lexical preprocessing

By restricting the results of the IR step to only those documents that contain relevant entities, the final dataset or corpus is ready for further processing. In this thesis, two lexical preprocessing techniques are often used: word stemming and entity blinding.

Stemming or lemmatization algorithms reduce words to their base form (stem or lemma), the primary lexical unit of a word. Stemming is usually implemented as a simple heuristic of removing inflectional forms, while lemmatization algorithms rely on dictionaries and morphological analysis. For example, the words 'interactions' and 'interacted' both have 'interact' as stem, but have two different lemmas: 'interaction' and 'interact' respectively. In textual analysis, a stemming or lemmatization step is usually applied to be able to group words with the same stem or lemma together, as these usually have similar meanings.

Blinding transforms a word into another word entirely and is often applied to the named entities in the text to facilitate information extraction. For example, all annotated proteins in a text mining corpus can be blinded with the string $protx$, as the relation extraction module is not concerned with the exact identity of each protein. On the contrary, it is easier to learn and recognise grammatical and lexical patterns such as 'transcription of $protx$', rather than creating distinct patterns for each possible protein name.

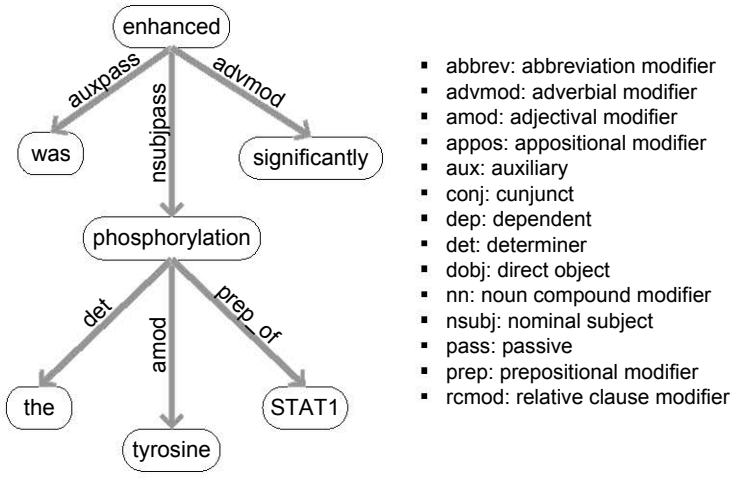


Figure 1.4: Dependency graph for the sentence 'The tyrosine phosphorylation of STAT1 was enhanced significantly'. Words of the sentence form the nodes of the graph, while edges denote their syntactic dependencies. Frequently occurring abbreviations are listed on the right (de Marneffe and Manning, 2011).

1.3.6 Syntactic analysis

Various syntactic analyses are often applied to determine the grammatical structure of a sentence by building the corresponding constituency tree or dependency graph.

A constituency tree relies on phrase chunking, which breaks a sentence down into noun, verb and prepositional phrases (constituents). These are then further broken down into even smaller constituents: part of speech (POS) tags. These annotations are produced by syntactic parsers and are important to help elucidate different word meanings (e.g. 'form' being either a noun or a verb). As depicted in Figure 1.3, the resulting parse tree represents the full syntax of a sentence.

Dependency parsing uses graph topology to represent grammatical relations (edges) between individual words (nodes) of the sentence. Dependency parsing is widely used for extracting relations from text, as it provides a compact and informative representation of the sentence structure. An exemplary dependency graph is depicted in Figure 1.4. Compared to the constituency tree, a dependency graph is much more compact and robust to syntactic variation.

$\{interaction\}$	:=	$[protein1]\ \{interact\_noun\}\ \{prep\}\ ([protein2] \mid \{list\});$
$\{interact\_noun\}$	:=	$association, colocalization, interaction, ...;$
$\{prep\}$	:=	$by, of, to, with, ...;$
$\{list\}$	:=	$([protein2],)?\ [protein2], ?\ and\ [protein2];$

Table 1.1: Example pattern in a rule-based system for extracting PPIs (Baumgartner *et al.*, 2008). The sentence '*TIMP-2 increases p27Kip1 association with Cdk4 and Cdk2*' would match this pattern twice, resulting in two predicted PPI pairs: (p27Kip1, Cdk4) and (p27Kip1, Cdk2).

1.3.7 Relation extraction

Relation extraction involves the automated extraction of structured information from text, in the form of relations between the (named) entities extracted previously. Roughly speaking, there are three broad categories of text mining algorithms for relation extraction: techniques based on lexical co-occurrence, rule-based approaches or machine learning techniques.

Co-occurrence methods are simple techniques that predict a relationship between two entities when they are mentioned in the same sentence, or when their co-occurrence in a document is statistically overrepresented. For these approaches, the application of NLP techniques is usually very minimal. Typically, such algorithms require manual filtering of the results as they exhibit high recall but low precision (Section 1.3.8).

A second important set of techniques apply **patterns or rules** which are usually hand-crafted and describe a certain template with fixed slots. The template can consist of both syntactic as well as lexical patterns. When a template can be matched against a certain sentence, the slots are filled with the entities of interest and a relation between them is predicted (Table 1.1). Rule-based algorithms usually obtain high precision while recall typically drops. This decrease in recall is due to the complex structure of English sentences, often containing multiple subordinate clauses. Interacting entities might then occur in text with some distance between them and can not be easily captured by the patterns. These issues can be solved in part by defining the patterns on top of dependency graphs (Section 1.3.6).

Finally, **machine learning** (ML) techniques are a wide variety of algorithms that aim at automatically elucidating complex properties from input data. Within the general class of ML techniques, a distinction is made between methods applied to labeled data or unlabeled data, termed supervised and unsupervised learning respectively. The most widely used unsupervised learning technique is clustering. Clustering algorithms group similar objects to find e.g. words with similar meanings in text (Section 5.3.1), or homologous sequences for defining gene families (Li *et al.*, 2003).

Within the field of BioNLP, many different supervised approaches are applied in a

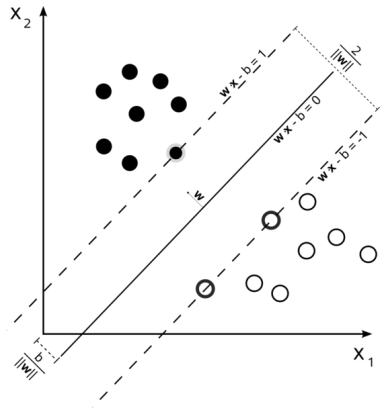


Figure 1.5: The maximum-margin separating hyperplane between two classes. Picture from Wikimedia Commons.

variety of applications. Decision trees are intuitive representations of a decision making process which, at each step, removes a subset of possible answers by evaluating a certain input variable and following the corresponding branches until a leaf node (target variable) is reached. Bayesian networks are based on graphical models and model the probabilistic relations between the input and output variables. A well known Bayesian model is the hidden markov model. Other widely used ML techniques are (artificial) neural networks, genetic programming and reinforcement learning. We refer to Larrañaga *et al.* (2006) for a general review on the application of machine learning techniques in bioinformatics.

In this thesis, we mainly focus on **support vector machines** (SVMs). SVMs are supervised ML methods often used in classification settings. They construct a model from labeled training data (instances) which can then be applied to a test set to predict the labels of new instances. In the context of BioNLP methods, instances may consist of a pair of proteins and the labels define the interaction type (e.g. PPI, regulation or none). The classification challenge requires the algorithm to recognise complex patterns in the training data and produce sufficiently general models that are capable of making predictions on unseen data. Meaningful features for the model construction are often derived from lexical properties of the sentence, as well as from constituency trees and dependency parses. Both global context, such as the size of the tree/graph, and local context, such as the children and ancestors of the entity-nodes, can be taken into account. Often, **feature selection** methods are used to select only a subset of automatically generated features, reducing noise for the classifier.

An SVM operates by defining a hyperplane that separates the different classes in the corpus, maximising the distance of that hyperplane to the nearest data points of each class (Figure 1.5). These samples on the margin are called the support vectors. We refer to Boser *et al.* (1992) for more details.

1.3.8 Performance measure

To assess the performance of any of the text mining sub-challenges, the metrics precision (p), recall (r) and F-measure (F) are often used. **Precision** measures the correctness of the predictions, usually expressed as a percentage. **Recall** or **sensitivity** on the other hand expresses how many of all possible correct answers were actually predicted. If we define A as the set of all possible correct results, and B as the set of all predictions, then

$$p = \frac{|A \cap B|}{|B|}$$

and

$$r = \frac{|A \cap B|}{|A|}$$

Alternatively, these metrics are defined by the number of true positives (tp), true negatives (tn), false positives (fp) and false negatives (fn), with *true* and *false* referring to the gold-standard labels of the test set, usually produced from manual annotations by experienced curators. The terms *positive* and *negative* refer to whether or not the classifier predictes a certain instance. The definitions then become:

$$p = \frac{tp}{tp + fp}$$

and

$$r = \frac{tp}{tp + fn}$$

There is a well-known trade-off between precision and recall (Buckland and Gey, 1994). Indeed, when applying stringent criteria (e.g. rule-based systems), typically high precision and low recall is achieved. When on the other hand predictions are made more loosely (e.g. co-occurrence based), higher recall will be obtained at the cost of low precision. As a consequence, these two metrics can not be used independently. Instead, the F_β**-score** is often used as the single evaluation metric and is defined as the harmonic mean between precision and recall:

$$F_\beta = \frac{(\beta^2 + 1) \cdot p \cdot r}{\beta^2 \cdot p + r}$$

Another widely used performance measure is **specificity** (s), measuring the ability to retrieve negatives:

$$s = \frac{tn}{tn + fp}$$

Specificity however is not commonly used in the context of NLP studies, as it focuses on the retrieval of negatives and any document contains virtually an infinite amount of statements that are *not* present in the text. Furthermore, the absence of a certain biological statement on e.g. a protein-protein interaction does not imply that interaction never happens, it just simply is not stated in that text. Therefore, NLP approaches mainly focus on positive instances using the measures precision, recall and F_β-score. Specifically, in this work we adopt the standard of using the F_1-measure and abbreviate it to simply F.

Finally, more detailed performance measures can be defined for any framework that ranks it predictions. For example, rule-based algorithms may assign a score to a prediction based on the confidence of a certain pattern and how well it matches. SVM-based classification algorithms can assign such scores by using the distance to the hyperplane as a measure for the confidence of the predicted label. For any such algorithm that ranks it predictions, it becomes possible to make the precision-recall trade-off explicit. Depending on the desired outcome, a confidence cut-off can be defined quite high or rather low, resulting in a small set of high-precision results or a larger set of high-recall predictions, respectively. Consequently, precision-recall curves are sometimes introduced to better visualise the possible performance results and trade-off parameters of a system.

1.4 Chapter overview

The next chapters of this thesis summarize my contributions to the field of BioNLP during my PhD studies. They are largely based on the peer-reviewed papers and manuscripts listed here.

Chapter 2 and 3

The next chapter discusses the development of a novel NLP classification framework based on machine learning techniques. We specifically detail the feature generation step which is at the core of the text mining classification algorithms presented in this thesis.

The third chapter discusses the application of this framework to the extraction of protein-protein interactions from text. It further details the necessity of a proper evaluation setup and provides guidelines for comparing different PPI extraction methods. Our methodology is compared against state-of-the-art techniques and benchmarked on

different corpora. Finally, we perform cross-corpus evaluations, illustrate the applicability of feature selection, and analyse the contribution of lexical and syntactic information.

These chapters are based on the following articles:

- **Van Landeghem, S.**, Saeys, Y., De Baets, B., Van de Peer, Y. (2008). Benchmarking machine learning techniques for the extraction of protein-protein interactions from text. *Benelearn*, p. 79-80. Spa, Belgium.
- **Van Landeghem, S.**, Saeys, Y., De Baets, B., Van de Peer, Y. (2008). Extracting protein-protein interactions from text using rich feature vectors and feature selection. *International Symposium on Semantic Mining in Biomedicine (SMBM)*, p. 77-84. Turku, Finland.

I contributed to these studies by performing an extensive literature review, designing the evaluation guidelines, implementing the NLP extraction framework for PPIs, performing the analyses and writing the manuscripts.

Chapter 4

This chapter discusses the extraction of biomolecular events from text by building upon the previously introduced classification framework. It further discusses our participation in the BioNLP 2009 Shared Task (ST), in which we obtained a 5^{th} place out of 24 international participants. Additionally, we have applied a novel technique of ensemble feature selection to the large-scale datasets, providing more cost-effective models and enhanced insight into the classification challenge. Finally, we have performed several interesting analyses such as precision-recall curves, learning curves and benchmarking of the different parameters of the framework.

It is based on the following articles:

- **Van Landeghem, S.**, Saeys, Y., De Baets, B., Van de Peer, Y. (2009). Analyzing text in search of biomolecular events: a high-precision machine learning framework. *BioNLP Shared Task Workshop*, p. 128-136. Colorado, USA.
- **Van Landeghem, S.**, De Baets, B., Van de Peer, Y., Saeys, Y. (2011). High-precision bio-molecular event extraction from text using parallel binary classifiers. *Computational Intelligence* 27(24), p. 645-664.
- **Van Landeghem, S.***, Abeel, T.*, Saeys, Y., Van de Peer, Y. (2010). Discriminative and informative features for biomolecular text mining with ensemble feature selection. *Bioinformatics* 26 (18): i554-i560.
  *: contributed equally

These studies were performed in collaboration with Dr. Thomas Abeel who designed and applied the ensemble feature selection analyses. I contributed by extending the text mining framework for event extraction, leading the team for participation in the ST'09, performing the analyses and writing the manuscripts.

Chapter 5

This chapter discusses the extraction of entity relations from text as a supporting task for event extraction. It further discusses our participation in the BioNLP 2011 Shared Task. Finally, we have combined our own framework (ranking 2^{nd}) with the winning system from Turku to analyse the performance discrepancy between the two.

It is based on the following articles:

- **Van Landeghem, S.**, Pyysalo, S., Ohta, T., Van de Peer, Y. (2010). Integration of static relations to enhance event extraction from text. *BioNLP Workshop*, p. 144-152. Uppsala, Sweden.
- **Van Landeghem, S.**, Abeel, T., De Baets, B., Van de Peer, Y. (2011). Detecting entity relations as a supporting task for bio-molecular event extraction. *BioNLP Shared Task Workshop*, p. 147-148. Oregon, USA.
- **Van Landeghem, S.**, Björne, J., Abeel, T., De Baets, B., Salakoski, T., Van de Peer, Y. (2012). Semantically linking molecular entities in literature through entity relationships. *BMC Bioinformatics* 13 (Suppl. 8): S6.

These studies were performed in collaboration with Dr. Sampo Pyysalo and Dr. Tomoko Ohta, who created the corpora. Further, Dr. Thomas Abeel performed the feature selection analysis. Finally, Jari Björne was responsible for generating the entity relation predictions with the Turku Event Extraction System (TEES).

I contributed by enhancing the text mining framework for the extraction of entity relations and integration with event extraction, leading the Ghent team for participation in the ST'11, combining the outputs of our own framework with TEES, performing the analyses and writing the manuscripts.

Chapter 6 and 7

The 6^{th} chapter discusses a large-scale text mining dataset, originally released by the University of Turku and covering the whole of PubMed, which we have extended by creating homology-based generalizations. Further, we transformed this textual resource into a MySQL database ('EVEX'). Additionally, an intuitive web interface was developed to enable manual exploration of the dataset. Finally, we have recently extended the

dataset by including full-text PubMed Central articles and providing integration with a state-of-the-art gene normalization system. The last chapter presents several interesting applications of EVEX, highlighting interesting directions for future work.

These chapters are based on the following articles and manuscripts:

- **Van Landeghem, S.**, Ginter, F., Van de Peer, Y., Salakoski, T. (2011). EVEX: A PubMed-scale resource for homology-based generalization of text mining predictions. *BioNLP Workshop*, p. 28-37. Portland, Oregon, USA.

- **Van Landeghem, S.**, Hakala, K., Rönnqvist, S., Salakoski, T., Van de Peer, Y. and Ginter, F. (2012). Exploring biomolecular literature with EVEX: Connecting genes through events, homology and indirect associations. *Advances in Bioinformatics*, in press.

- Kaewphan, S., Peltonen, S., **Van Landeghem, S.**, Van de Peer, Y., Jones, P., Ginter, F. (2012). Integrating large-scale text mining and co-expression networks: Targeting NADP(H) metabolism in *E. coli* with event extraction. *LREC workshop on Building and Evaluating Resources for Biomedical Text Mining*, in press.

- Björne, J.*, **Van Landeghem, S.***, Wei, C.H., Hakala, K., Pyysalo, S., Ananiadou, S., Kao, H.Y., Lu, Z., Salakoski, T., Van de Peer, Y. and Ginter, F. (under review since April. 2012). Bibliome-wide event extraction and integration with biomolecular database records across the taxonomic space. Submitted to *Bioinformatics*.
  *: contributed equally

EVEX was designed and implemented in close collaboration with Dr. Filip Ginter, with significant contributions by Jari Björne (system development), Sampo Pyysalo (analyses and algorithms), Kai Hakala (website development), Chih-Hsuan Wei (gene normalization) and Zuzanna Drebert (manual evaluations). The study on NADP(H) metabolism in *E. coli* was mainly conducted by Suwisa Kaewphan and Sanna Peltonen.

I contributed to the implementation of the family-based disambiguation algorithm, the design and implementation of the database, the design of the website, running the gene normalization algorithm, providing support for the *E. coli* use case, performing various analyses and writing the manuscripts.

2 NLP framework

Text mining tools have become a necessity to keep up with the ever increasing pace of publications in the field of molecular biology. By extracting and summarizing biological knowledge from unstructured articles, text mining enables large-scale analysis and full integration of knowledge stored in both databases and research articles.

This thesis presents several state-of-the-art algorithms contributing to this goal, based on machine learning (ML) techniques (Section 1.3.7). This chapter discusses specifically the basic principles of the feature generation module of the ML framework applied in the next three chapters, as this module is an essential component at the transition of unstructured text to a structured knowledge representation.

2.1 Classification model

Our supervised-learning framework is based on support vector machines (SVMs) (Boser *et al.*, 1992). The SVM is a data-driven method for solving two-class classification tasks, based on the concept of large margins, and is known to perform well in high-dimensional spaces (Saeys *et al.*, 2007). Specifically, we use the Weka[1] implementation of LibSVM. The choice of kernel and parameter tuning is discussed within the context of the specific extraction challenges as described in the next chapters.

[1] Available at `http://www.cs.waikato.ac.nz/ml/weka/`

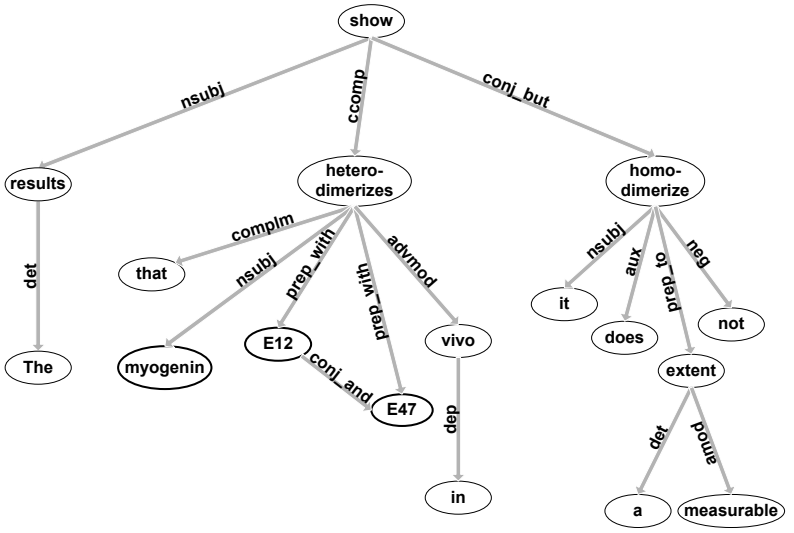


Figure 2.1: The dependency graph for the sentence 'The results show that myogenin heterodimerizes with E12 and E47 in vivo, but it does not homodimerize to a measurable extent'.

2.2 Dependency parsing

For each sentence in the NLP corpus, a constituency tree (Section 1.3.6) is built using the Stanford parser (de Marneffe *et al.*, 2006), providing phrase structures and part-of-speech tags. Further, this tree is converted to a dependency graph, selecting the option for collapsed dependencies with propagation of conjunct dependencies. The resulting dependency structures are directed graphs (Figure 2.1).

Rather than consulting the full dependency graph of a sentence, sub-graphs are first defined by selecting all relevant nodes for a certain relation (instance). For example, when analysing the relationship between *myogenin* and *E12*, the sub-graph with the root node *heterodimerizes* is selected, and all other dependencies discarded. This prevents the inclusion of irrelevant features.

2.3 Feature generation

To capture the semantics within a sentence and between two biomolecular entities of interest, several different features are extracted for inclusion in the feature vector, partly based on previous work by Kim *et al.* (2008b). To improve generalization of lexical

information by the classifier, we apply the Porter stemming algorithm to all lexical features (Porter, 1980). This algorithm maps words to their stem by applying a suffix-stripping algorithm (Section 1.3.5).

2.3.1 Vertex walks

Vertex walks or v-walks are patterns derived from the dependency graph, by combining two subsequent vertices (nodes) with their intermediate edge. V-walk features are included in the feature vector in two variants: the lexical variant includes the stemmed, lexical information for the nodes, while the syntactic variant only considers the part-of-speech tags of these nodes. As an example, the relation between *myogenin* and *E12*, as analysed on the dependency graph depicted in Figure 2.1, results in the patterns 'heterodimer nsubj myogenin', 'heterodimer prep_with E12', 'VBZ nsubj NN' and 'VBZ prep_with NN'.

2.3.2 Edge walks

Edge walks or e-walks are similar to v-walks, but contain information on two subsequent edges and their common vertex (*e-walk*). Again considering the relation between *myogenin* and *E12*, the e-walk features would include 'nsubj heterodimer prep' (lexical variant) and 'nsubj VBZ prep' (syntactic variant).

2.3.3 Bag-of-words

A bag-of-words (BOW) approach represents a certain text as an unordened collection of (stemmed) words. This technique can be applied to the whole sentence and may or may not exclude uninformative word classes such as prepositions. However, a BOW approach might still result in many irrelevant features such as those extracted from uninformative sub-sentences. Alternatively, only the sub-sentence spanning the actual event in text could be considered. A final option is to only include the words of the nodes spanning the relevant dependency sub-graph, including only highly relevant lexical information.

2.3.4 N-grams

N-grams are formed by listing N consecutive (stemmed) words from the sentence or sub-sentence relevant to a given relation. BOW features can be seen as 1-grams or unigrams, and other common values of N are 2 (bigrams) and 3 (trigrams).

2.3.5 Additional features

Additional features highly depend on the nature of the relation extraction challenge, and may include for example the size of the dependency graph or sub-graph, as well as information on the root node or specific interaction words in the sentence (Section 4.2.2).

2.4 Feature blinding

Generalizability of the classifier is a crucial aspect for any text mining framework that needs to be able to extract events concerning previously unsequenced or unpublished genes. As such, blinding techniques can be applied to all previously described features. For example, protein names can be blinded, i.e. substituted by the token *PROTX*, to enable the classifier to learn general interaction patterns, disregarding the specific proteins involved. Other examples may include blinding the interaction word with its event type (e.g. 'heterodimerizes' becomes *BINDX*). Rather then only blinding the lexical content of the word, an artificial part-of-speech tag may also be used to provide generalization of the syntactic variants of the patterns described in the previous sections.

Blinding of features could also incorporate domain information, using for example annotated GO terms for a certain protein (Section 1.2.2) or other methods to categorize molecular entities into functional clusters and using the cluster labels for blinding them. These opportunities are not further considered for protein or gene names in this work, as they would bias the results towards known information. However, we do apply this technique to categorize domain terms ('promoter', 'complex') into semantic classes, as described in Section 5.3.2.

When blinding is applied to certain features, there is the option of having both a blinded and a non-blinded variant in the feature vector. However, in this work, protein names are always blinded, again to avoid extraction bias to well-studied proteins. Their part-of-speech tags are always blinded as well, as this provides valuable information to the classifier.

2.5 Feature encoding

All features are encoded in the feature vector by assigning a numeric value for each syntactic or lexical pattern, expressing the number of times that pattern occurs in the sentence or its derived dependency graph. This encoding technique results in sparse feature vectors and high-dimensional feature sets, creating a need for feature selection techniques (Section 3.2.3 and Section 4.4).

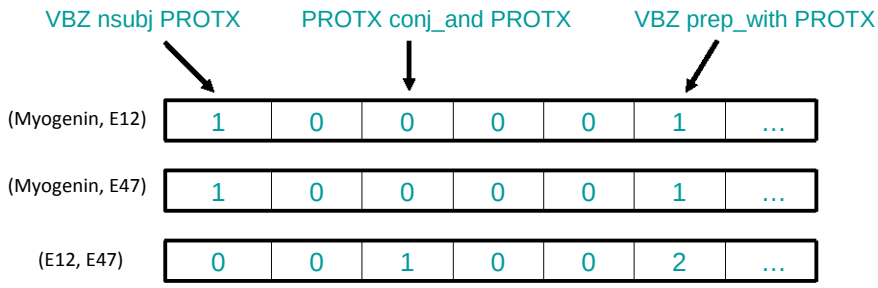


Figure 2.2: Example of feature encoding for syntactic vertex walks, as derived from Figure 2.1 for different protein pairs.

Figure 2.2 depicts a representative example of a part of a feature vector, showing the blinded syntactic vertex walks for different protein pairs as extracted from the dependency sub-graph on 'heterodimerization' (Figure 2.1). The first two vectors, representing the pairs *Myogenin-E12* and *Myogenin-E47* respectively, correspond to positive instances, while the third pair (*E12-E47*) is a negative instance and shows highly different features. Note that, when only considering the shortest path between *E12* and *E47*, the node 'heterodimer' would not be included in the sub-graph and the 'VBZ prep_with PROTX' feature would be 0.

3 Protein-protein interactions

This chapter describes various corpora and algorithms for the extraction of protein-protein interactions (PPIs) from text. Further, a thorough analysis of how to define a meaningful evaluation framework for the PPI extraction challenge is presented (Section 3.1). The development of a novel machine learning framework is described (Section 3.2), and evaluated on various datasets (Section 3.3). Finally, we present a feature selection study, obtaining more cost-effective models and analysing the contribution of lexical and syntactic information.

Related work

The extraction of protein-protein interactions from research articles has attracted wide interest in the field of BioNLP. However, the definition of PPI varies greatly across studies. A strict interpretation includes only non-generic, physical protein-protein interactions. Within the field of BioNLP, PPI datasets and algorithms sometimes additionally contain protein-DNA interactions, chemical modifications or genetic relations (Section 1.1.3). These variations depend on the specific PPI text mining corpus and its annotation guidelines (Section 3.1.1).

Many different methodologies have been proposed for the extraction of PPIs from text. The first category of methods is based on co-occurrence, classifying two proteins as interacting when mentioned in the same sentence, or when their co-occurrence in an

abstract is statistically overrepresented (Ding *et al.*, 2002; Rebholz-Schuhmann *et al.*, 2007).

A second important set of techniques applies (hand-crafted) patterns or rules expressing the required syntactic structure of the sentence. Sometimes, the patterns are not matched to the sentence, but to the corresponding dependency graph (Section 1.3.6). The RelEx system, for example, uses three rules in combination with information derived from dependency graphs (Fundel *et al.*, 2006).

Finally, machine learning techniques have been widely applied for the extraction of PPIs, often including both lexical and syntactic information, originating from the sentence itself and its dependency graph (Katrenko and Adriaans, 2007; Erkan *et al.*, 2007; Kim *et al.*, 2008b; Saetre *et al.*, 2008). A more reduced feature set is used by Fayruzov *et al.* (2008), taking mainly syntactic information into account. A hybrid approach is also possible, with hand-crafted rules forming the basis for different kernels, which are then aggregated by linear combination (Giuliano *et al.*, 2006).

3.1 Evaluation framework

While studying state-of-the-art systems that extract PPIs from text, it became clear that this field is struggling with a heterogeneous collection of datasets and evaluation methods. We first analyse these problems to determine a meaningful evaluation framework for this study.

3.1.1 PPI corpora

A comparative study between different PPI extraction systems is a non-trivial task, as different studies often benchmark on different datasets. Relevant PPI corpora include AIMed (Bunescu *et al.*, 2005), BioInfer (Pyysalo *et al.*, 2007), HPRD50 (Fundel *et al.*, 2006), IEPA (Ding *et al.*, 2002) and LLL (Nedellec, 2006). These corpora all have slightly different scopes, ranging from protein-gene interactions concerned with *Basilicus subtilis* transcription to human protein-protein interactions. It has been shown that the choice of benchmark dataset dramatically influences extraction performance. For example, the RelEx system of Fundel *et al.* (2006) has been reimplemented with the goal of evaluating it on different corpora (Pyysalo *et al.*, 2008). An F-score of 77% was obtained when benchmarking on LLL, and a score between 41% and 44% when evaluated on AImed and BioInfer. We obtain similar results when applying the walk kernel of Kim *et al.* (2008b) to the AImed dataset, which results in an F-score of 44%. In contrast, the original paper reports a score of 77% for the evaluation on LLL. This shows

that for the same extraction method, performance can differ up to 36 percentage points (pp) depending on the choice of the corpus.

To enable meaningful comparisons between various information extraction techniques, tools have been released to convert these different datasets into a common data format (Pyysalo *et al.*, 2008). Another important resource is the BioCreative initiative, which aims to provide a framework for the construction of suitable 'gold-standard' datasets, applicable for text mining systems in biology (Hirschman *et al.*, 2005). Finally, the GENIA corpus can be useful for benchmarking various subtasks of text mining algorithms (Kim *et al.*, 2008a).

3.1.2 The extraction task

The formal representation of the PPI extraction task is not unambiguously defined across the different corpora. The LLL dataset and BioInfer both consider the semantic role of the different proteins in the interaction and discriminate between effectors (causes) and effectees (themes). In AImed however, protein-protein interactions are considered to be symmetrical. This has led to the common practice of treating the annotations in all corpora as symmetrical, resulting in artificially higher precision rates.

To enable relation extraction, it is necessary to first locate the named entities in text, i.e. protein mentions (Section 1.3.3). When performing the named entity recognition (NER) step automatically, errors will propagate and cause a drop in performance. However, we believe that the NER step is a sub-challenge in its own right and should be examined and evaluated separately. For this reason, we follow the common practice of assuming known gold-standard entities in all datasets.

3.1.3 Instance creation

Even when evaluating on the same dataset, different preprocessing steps can yield a varying set of instances. First of all, most methods only consider PPI annotations when stated within a single sentence. Further, homodimers, which are self-interacting proteins, are sometimes discarded from the dataset. Similarly, nested or overlapping gene/protein mentions in the corpus are sometimes not processed correctly, influencing the final number of instances in the dataset and, ultimately, the global performance of the system.

Finally, most corpora do not deal with the construction of negative training data. As a consequence, it has become common practice to adopt the closed world assumption, stating that no interaction exists between two entities when there is no annotated evidence. Even though AImed provides an explicit set of abstracts with no annotated interactions, these are not always used, resulting in a varying number of negative instances in the training set.

3.1.4 Counting true positives

The definition of a true positive is ambiguous in the text mining domain. Each pair of proteins in a sentence is usually considered as an individual instance, evaluated independently of others. Some however state that an interaction between two proteins may be expressed in the same corpus by more than one instance. Because it suffices to extract only one instance for each true interaction, the latter evaluation technique exhibits higher recall.

In this thesis, we employ the first model (instance-level evaluation) when developing and evaluating natural language processing techniques (Chapters 3, 4 and 5), but resort to the second model (corpus-level evaluation) when evaluating large-scale text mining datasets developed for practical use cases (Chapter 6 and 7).

3.1.5 Cross-validation

Finally, the evaluation setup can vary significantly, yielding uncomparable performance figures. In most studies, K-fold cross-validation (CV) is used, dividing the dataset into K subsets and repeating the evaluation K times with one of the subsets excluded from the training set and used for testing. Typical values of K are 5 and 10.

When performing CV in an ideal setting, abstracts for the testing phase are completely hidden during training. However, some evaluations exhibit an artificial boost of performance by using features from the same sentence in both training and testing steps of the machine learning process (Saetre *et al.*, 2008). This effect is caused by the fact that one sentence in the dataset yields C_n^2 distinct instances, where n is the number of proteins in the sentence and each instance represents a pairwise combination of proteins.

3.2 Classification framework

In our study, we use all the datasets that have been converted to the common data format by Pyysalo *et al.* (2008), with the exception of BioInfer. The latter corpus contains extensive annotations of proteins and interactions that are not compatible with our extraction method. For example, the words *alpha 5 integrins* are annotated as being a protein reference in the construct *alpha 5 and beta 1 integrins*. However, our extraction method assumes a protein is mentioned as a contiguous stream of tokens. This is why we exclude BioInfer from further analysis and focus on the other four corpora: AIMed, HPRD5, IEPA and LLL.

Dataset	Positive	Negative	Total
AIMed	1000	4670	5670
HPRD50	163	270	433
IEPA	335	482	817
LLL	164	166	330
All	1662	5588	7250

Table 3.1: Number of instances in the four corpora.

3.2.1 Dataset preprocessing

We use the gold-standard protein annotations which are available for all corpora. As not all datasets provide annotation of the direction of interactions, we consider interactions to be symmetric (Section 3.1.2).

In preparing the datasets we excluded homodimers, as not all corpora support homodimer annotation (Section 3.1.3). Sentences with at least two co-occurring proteins are selected for further processing, creating a distinct instance in the dataset for each pairwise combination of proteins in the sentence. Nested annotations are taken into consideration in all datasets. We apply the closed-world assumption to create negative instances, assuming there is no interaction between two proteins when there is no annotated evidence. For AIMed, the abstracts included in the corpus that contain no interactions are also taken into account. The resulting numbers of positive and negative instances are shown in Table 3.1.

3.2.2 Classification

For classification, we apply a linear support vector machine (Section 2.1) with an internal 5-fold cross-validation loop on the training portion of the data to determine the optimal c-parameter. The c-parameter of an SVM model is the cost parameter, defining the trade-off between training errors and model complexity.

Our feature extraction method extracts useful patterns derived from the shortest path between two proteins in the dependency graph, including vertex-walks (Section 2.3.1) and edge-walks (Section 2.3.2). Further, a bag-of-words approach is applied to the full sentence (Section 2.3.3), giving rise to quite some irrelevant features. This will be one of the main motivations to apply fully automated feature selection techniques after feature extraction (Section 3.2.3). Finally, syntactic and lexical information from the root node, as well as the length of the shortest dependency path, are stored as additional features.

The protein names of the two proteins under consideration for PPI extraction, are blinded as described in Section 2.4. Table 3.2 summarizes the features extracted for the

Type	Features
Lex v-walk	heterodimer nsubj PROTX, heterodimer prep PROTX
Syn v-walk	VBZ nsubj PROTX, VBZ prep PROTX
Lex e-walk	nsubj heterodimer prep
Syn e-walk	nsubj VBZ prep
BOW	PROTX, a, and, but, doe, extent, heterodimer, homodimer, in, it, measur, not, result, show, that, the, to, vivo, with
Lex root	heterodimer
Syn root	VBZ

Table 3.2: Syntactic and lexical features for the pair of proteins (*Myogenin*, *E12*) from Figure 2.1.

pair of proteins (*Myogenin*, *E12*) as analysed from the dependency graph depicted in Figure 2.1 (page 2-2).

3.2.3 Feature selection

Feature selection (FS) methods are a class of dimensionality-reduction techniques that aim at identifying a subset of the most relevant features from a potentially large initial set of features. In contrast to other reduction techniques such as methods based on projection, FS only selects a subset of the original set of features, preserving the original semantics.

Advantages of applying feature selection include its potential to improve generalization performance by avoiding overfitting, faster and more cost-effective models and gaining a deeper insight into the underlying processes that generated the data. Depending on the interaction with the model, three classes of FS techniques can be defined (Guyon and Elisseeff, 2003). In this work, we focus on the class of *filter* methods, which perform feature selection by looking only at the intrinsic properties of the data, thus being independent of the classification model used afterwards. Advantages of this class of methods include their scalability to high-dimensional datasets (such as the ones we deal with in this work) and their speed. An in-depth analysis of the different classes of FS techniques, as well as their application in bioinformatics, can be found in Saeys *et al.* (2007).

The application of FS in the domain of natural language processing is relatively new. Previous studies were mainly focused on feature type selection, investigating the type of features that are potentially useful for relation extraction (Jiang and Zhai, 2007). Feature selection techniques have further been employed for the task of text classification (Wang *et al.*, 2008). However, to the best of our knowledge, we have presented the first study of applying rich feature vectors in combination with feature selection for protein-protein

interaction extraction (Van Landeghem *et al.*, 2008). Our study using fully automated feature selection methods is clearly different to previous work concerning manually selected varying sets of features (Katrenko and Adriaans, 2007).

The filter method used in this work is based on the information-theoretic concept of *gain ratio*. A given set of training patterns S can be regarded as a distribution over the class labels, and its entropy can be calculated as

$$H(S) = -\sum_{i=1}^{s} p(c_i) \log_2 p(c_i)$$

where $p(c_i)$ denotes the proportion of patterns in S belonging to class c_i. The *information gain* $IG(S, D)$ then represents the expected reduction in entropy (uncertainty) when splitting on a feature D, and can be calculated as

$$\begin{aligned} IG(S, D) &= H(S) - H(S|D) \\ &= H(S) - \sum_{j \in V(D)} \frac{|S_j|}{|S|} H(S_j) \end{aligned}$$

where $V(D)$ denotes the possible values for feature D and S_j is the subset of S for which feature D has value j.

To adjust the bias towards features with a larger number of possible values, the information gain should be scaled by the entropy of S with respect to the values of feature D, resulting in the *gain ratio* $GR(S, D)$:

$$GR(S, D) = \frac{IG(S, D)}{-\sum_{j \in V(D)} \frac{|S_j|}{|S|} \log_2 \frac{|S_j|}{|S|}}$$

Applying the gain ratio to every feature in the dataset gives an estimate of the feature's importance. Consequently, all features can be ranked from most influential to least influential by sorting their gain ratios. The top k features can then be used to construct a simplified classifier.

3.2.4 Evaluation strategy

For benchmarking our PPI extraction method, we use instance-level evaluation (Section 3.1.4). We have applied regular 10-fold cross-validation (*Instance CV*), as well as a modified version of 10-fold cross-validation (*Abstract CV*), with folds consisting of complete abstracts (Section 3.1.5). As a performance measure, we adopt the commonly used F-measure.

In addition to training and testing on a single dataset using CV, we have conducted a large-scale evaluation using all four corpora. The rationale for this approach was to analyse the scalability of our approach. Most datasets have been constructed using specific keywords (e.g. LLL: *Bacillus subtilis transcription*), which causes a bias in the

	Corpus	p	r	F
Inst. CV	AIMed	66	58	62
	HPRD50	71	71	71
	IEPA	74	69	71
	LLL	79	84	**82**
Abstr. CV	AIMed	49	44	46
	HPRD50	60	51	55
	IEPA	64	70	67
	LLL	72	73	**73**
Co-occ.	AIMed	18	100	30
	HPRD50	38	100	55
	IEPA	41	100	58
	LLL	50	100	**66**

Table 3.3: Evaluation on the four individual datasets.

classifier towards this particular domain. However, when using features from three different datasets and testing on an independent dataset, we obtain a more diverse model, which is more representative for the real-world task of extracting interactions from various PubMed abstracts. We conducted four experiments, each time using a different corpus as test set, while including the other three in the training data. To the best of our knowledge, our study was the first one to include such large-scale cross-dataset comparisons (Van Landeghem *et al.*, 2008).

3.3 Results

3.3.1 Individual dataset evaluation

We have evaluated our method on all datasets separately, using both 10-fold instance CV and abstract CV (Table 3.3). For the evaluation on AIMed, the original abstract splits were used (Bunescu *et al.*, 2005). We notice an artificial boost of performance of up to 16 percentage points (pp) in F-measure when using instance CV.

In both experiments we also find a significant difference in F-measure between the best results (LLL) and the worst (AIMed), ranging between 20 and 27 pp. To demonstrate the inherent differences between the four individual datasets, we have included the results of a simple co-occurrence based technique, assigning a true interaction between each co-occurring pair of proteins. These results exhibit a difference in F-measure of up to 36 pp. between AIMed and LLL. Obviously, extracting relations from LLL should be significantly easier, as two out of 3 co-occurring protein pairs interact, compared to less than 1 out of 3 in the AIMed corpus. A large part of these differences can be contributed to the annotation guidelines of the corpora (Pyysalo *et al.*, 2008).

		Method	p	r	F
AIMed	(abstr. cv)	Van Landeghem *et al.* (2008)	49	44	46
		Fundel *et al.* (2006)	40	50	44
		Giuliano *et al.* (2006)	61	57	**59**
		Saetre *et al.* (2008)	64	44	52
AIMed	(inst. cv)	Van Landeghem *et al.* (2008)	66	58	62
		Erkan *et al.* (2007)	60	61	60
		Fayruzov *et al.* (2008)	41	50	45
		Katrenko and Adriaans (2007)	45	68	54
		Saetre *et al.* (2008)	78	63	**70**
LLL	(inst. cv)	Van Landeghem *et al.* (2008)	79	84	**82**
		Fayruzov *et al.* (2008)	72	86	78
		Fundel *et al.* (2006)	85	79	**82**

Table 3.4: Comparison of our method to existing techniques.

We further compared our method to other PPI extraction techniques. To allow for a fair comparison, we only considered studies using a similar task definition and evaluation setup (Table 3.4). We observe that our method is comparable to state-of-the-art performance, and that it achieves particularly good results when using instance CV on the LLL dataset.

3.3.2 Cross-dataset experiments

To assess the performance of our method in a more realistic setup, we have conducted large-scale cross-dataset experiments. For this purpose, we used one dataset for testing, and the other three for training, limiting the training bias. These experiments provide an estimate of the out-of-domain generalizability of the classifier, by analysing the artificial boost in performance when only performing a single-domain evaluation.

The results of our experiments are shown in Table 3.5 (rows 'all'). We see that testing on HPRD50 achieves the best performance, with 62% precision, 52% recall and 57% F-measure. For this corpus, the performance is similar to the single-dataset evaluation (Table 3.3, 'Abstr. CV'). However, we observe a large drop in performance when testing on IEPA and LLL, and to a smaller extent, on AIMed. This shows that studies using single-dataset evaluations are biased towards the specific properties of the corpus used. It confirms the need for extrinsic evaluations of text mining tools as stated by Caporaso *et al.* (2008).

3.3.3 Feature selection

Because our extraction method results in high-dimensional, sparse feature vectors, we have investigated the applicability of feature selection techniques to improve accuracy

Test set	Features	p	r	F
AIMed	all	27	67	**38**
	syntactic	28	58	37
	lexical	24	72	36
HPRD50	all	62	52	**57**
	syntactic	70	48	**57**
	lexical	60	50	54
IEPA	all	87	27	**41**
	syntactic	62	26	37
	lexical	82	17	29
LLL	all	54	32	40
	syntactic	64	30	**41**
	lexical	47	28	35

Table 3.5: Cross-dataset experiments using lexical information, syntactic information or both.

and obtain faster models. The results of these experiments on the individual datasets are shown in Table 3.6. On HPRD50, recall could be increased with 11 pp., resulting in an increase in F-measure of 6 pp., while less than 20% of the features were kept. For IEPA and LLL, F-measure remains stable when using respectively 36% and 25% of all available features. These results indicate that FS can reduce the feature set considerably without loss of performance. For the more extensive dataset AIMed, the number of extracted features and training instances are multiplied by a factor 10 in comparison to the other datasets, which induces greater complexity. On AIMed, we can filter out 29% of all features while still obtaining the same performance. If we filter out 64%, keeping only 5000 features of the original set, the F-measure drops with 5 pp. However, the time necessary to build the classifier for all ten folds is reduced from 365 minutes to 202 minutes, including the FS step itself. These results clearly illustrate the usefulness of feature selection to create more cost-effective models.

The cross-dataset experiments give rise to even more high-dimensional datasets, with up to 26.700 features. Applying FS on these experiments results in similar findings, reducing the feature set significantly (up to 50%) without loss of performance (data not shown). This again shows how, through FS, we can obtain faster models with less risk of overfitting.

3.3.4 Lexical vs. syntactic information

To gain deeper insight into the importance of certain feature types, we performed a statistical analysis of their contribution before and after FS. In general, the percentage of syntactic features rises after filtering, usually accompanied by a reduction of BOW features (Table 3.6, last three columns). However, lexical information from v-walks and e-walks still takes up the biggest part of the feature set.

	Features	p	r	F	syn	lex	bow
AIMed	14.000	49	44	**46**	15	61	20
	10.000	48	43	45	16	61	19
	7.500	41	41	41	17	61	18
	5.000	44	38	41	16	59	21
HPRD50	2.600	60	51	55	21	44	29
	1.500	51	60	55	23	48	23
	750	57	61	59	23	52	20
	500	61	62	**61**	23	45	28
	250	58	36	45	23	51	23
IEPA	6.900	64	70	67	17	49	30
	5.000	61	71	65	14	43	38
	2.500	63	75	**68**	22	51	21
	1.000	54	66	60	20	42	34
LLL	1.600	72	73	**73**	22	44	28
	800	75	71	**73**	27	48	19
	400	68	77	**73**	33	44	18
	200	54	66	60	35	58	3

Table 3.6: FS on individual datasets, showing the distribution of the three most important type of features in percentages (syntactic walks, lexical walks and BOW-features). The table excludes other less abundant features such as information on the root node of the dependency graph. The evaluation is performed using Abstract CV.

To test the hypothesis that both lexical and syntactic information are important when extracting protein-protein interactions, we have re-run the cross-dataset experiments once with only lexical information, and once with only syntactic information. The results are shown in Table 3.5, demonstrating that the global performance of both lexical and syntactic approaches are comparable for most experiments. However, when using only syntactic information and comparing this approach to the full feature set, a gain of precision of up to 10 pp. can be achieved, while producing a similar F-score.

A notable exception to these general rules is when IEPA is used as testing set. In this particular case, high precision is achieved by mainly lexical information, while the F-score of lexical-only information is significantly lower than the syntactic-only approach.

In general, however, it is clear that a purely syntactic approach can produce satisfying performance, while using only 10-15% of the original feature set. These results support the hypothesis that using only syntactic information leads to classifiers that are able to perform well, while being independent of a specific lexicon (Fayruzov *et al.*, 2008). However, to improve recall, including lexical information might still be useful.

3.4 Discussion and conclusion

The comparison of different PPI extraction methods is hindered by the lack of standard evaluation procedures. We have discussed important issues for such a comparative study and indicated practical guidelines for setting up a meaningful evaluation.

Further, we have developed a novel machine learning technique to extract protein-protein interactions using rich feature vectors. For the extraction of relevant features, syntactic and grammatical information from dependency/constituency parsing was used, as well as lexical information from the sentence. As an important novelty, we have conducted cross-dataset experiments which offer a more realistic view on the performance of our method. Finally, for the first time in this domain, we have applied feature selection techniques to show these can lead to faster and more cost-effective models. Analysing the feature sets from our experiments before and after feature selection, we have shown the importance of combining both lexical and syntactic information for the extraction of interactions from text.

During evaluation of our PPI methods, it has become clear that the various PPI corpora exhibit intrinsic differences and that their annotation guidelines sometimes result in artificial high performance rates. For example, the LLL dataset excludes sentences without interactions and does not annotate non-interacting proteins, resulting in a significant bias towards correct information with 2 out of 3 co-occurring protein pairs annotated as interacting. In the next chapter we discuss biomolecular event extraction from text in the framework of a community-wide challenge, solving comparability issues and relying on a much more extensive text mining corpus.

4

Event extraction

Shifting focus from binary relation extraction such as protein-protein interactions, a more extensive extraction challenge was popularized by the community-wide *BioNLP Shared Task on Event Extraction* (Kim *et al.*, 2009, 2011). The goal of this challenge is to reliably extract several biological events from text. These events concern protein metabolism (e.g. transcription and catabolism), protein modification (e.g. phosphorylation) and fundamental molecular events (e.g. binding and localization). Furthermore, causal relations are represented by specific regulatory events, providing the opportunity to model more complex pathways than ever before.

In this chapter, we present an adaptation of our PPI-extraction framework described in Chapter 3. We have refined and extended this machine learning framework to be able to extract complex event structures, carefully benchmarking the influence of various design choices (Section 4.2). The resulting framework was applied for participation in the BioNLP Shared Task of 2009, obtaining a 5^{th} place out of 24 international participants (Section 4.3).

Building further on the encouraging results of using FS for PPI extraction, Section 4.4 presents a more advanced FS methodology which combines multiple weak feature selectors into a single robust one. This ensemble feature method is applied to the event data to gain a better insight into the black-box classification algorithm and to create more cost-effective models.

4.1 Extraction challenge

In the context of the BioNLP Shared Task 2009 (ST'09), 9 distinct event types are defined. The various event types were selected from the GENIA ontology (Kim *et al.*, 2008a) and represent some of the most important events in protein biology.

4.1.1 Physical event types

Covering protein metabolism, protein modification and fundamental molecular events, 6 event types are denoted as 'physical' event types as they involve specific physical interactions or pathways (Kim *et al.*, 2009).

Gene expression refers to the process of transferring the DNA code to an RNA strand (transcription) and translating this sequence to a protein (Section 1.1.2). These events are often stated in research articles describing a relation between expression levels and observed phenotypes, e.g. 'Certain mec1 mutations or *overexpression of Mec1p* lead to shortened telomeres and loss of telomeric silencing'.

Transcription is in fact part of the general process of gene expression, but is treated as a separate event type in the ST corpus. Statements involving transcription often specifically mention an activator or inhibitor of the transcribed gene: 'In pregnant ewes, IFNT inhibits *transcription of the ESR1 gene*'.

Protein **phosphorylation** denotes the physical modification of a protein by a protein kinase (Section 1.1.3) and is the most widely studied post-translational modification. Phosphorylation is known to be crucial for controlling the behavior of a protein and thus regulating various cellular processes. For these events, the ST corpora additionally include the specific site argument at which the protein is phosphorylated, e.g. 'in vitro kinase assays revealed that *p53 was phosphorylated* at Ser46 by ATM-AS'. Dephosphorylation is not annotated in the corpus.

Protein **localization** captures the presence of a protein, or a change in its location. Additional arguments specify the current and/or target location. For example: '*Mec1 and Ddc2 localize* to sites of DNA damage by interacting with RPA in the form of the Mec1-Ddc2 complex'.

Protein catabolism is concerned with the breakdown of proteins into their individual amino acids. While complex pathways underly this process, it is usually presented in text using general, high-level descriptions, e.g. 'The *degradation of CycA* is delayed in response to DNA damage'.

Finally, **binding** events include various biomolecular interactions, including protein-DNA binding, protein-protein interactions and complex formations of more than 2 subunits (Section 1.1.3). Due to insufficient coverage in the training data, we do not consider binding events of more than 2 arguments. Within the ST corpus, binding events may contain specific binding sites, e.g. 'the S392E mutation does not increase *p53 binding* to its 20 bp consensus DNA sequence in the absence of nonspecific DNA additives'.

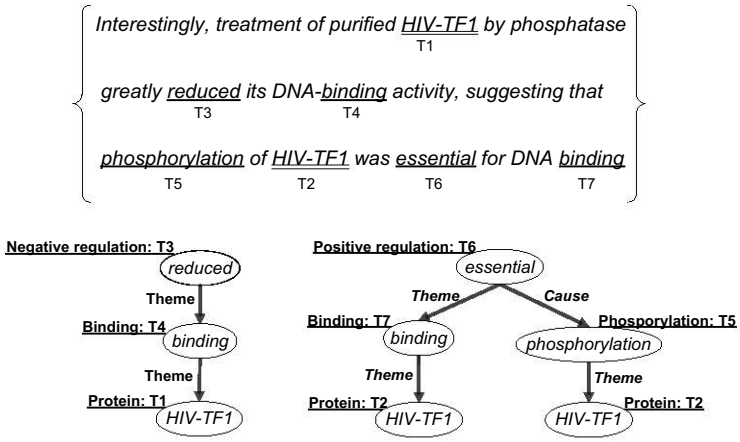


Figure 4.1: An example sentence from PubMed article 1653950 of the training corpus. It contains 5 events: two (single-argument) binding events, one phosphorylation, one negative (single-argument) regulation and one positive (double-argument) regulation event. GGPs and triggers are marked and assigned a unique identifier.

4.1.2 Regulatory event types

Additionally, three regulatory event types are defined: unspecified regulation, positive regulation and negative regulation. These event types cover up- and downregulation of RNAs and proteins, as well as identifying specific causal relations for some of the event types described previously. For example, '*phosphorylation of p53 by ATM-AS*' involves a phosphorylation event of *p53*, which as a whole is the target for a positive regulation caused by *ATM-AS*. Such recursive event definitions enable a complex model of regulatory pathways concerning various physical events. Figure 4.1 shows a representative example of the complexity of this task, depicting two regulatory events that have binding and phosphorylation events as their arguments.

4.1.3 Formal representation

In the ST corpus, each event is characterized by a trigger, such as 'heterodimerization' for a binding event. Such triggers are linked to a set of gene/protein symbols to define the full event. The corpus makes no distinction between gene or protein names, as they are often used interchangeably. We thus refer to all genes and gene products as 'GGPs'. The GGP-argument of an event which undergoes the modification, is denoted as the '(T)heme' (affectee), while the (optional) '(C)ause' (affector) argument drives the action. Further, regulatory events can have either GGPs or other events as arguments.

Event type	Args	Train	Devel	Test
Protein catabolism	T	110	21	14
Phosphorylation	T	169	47	139
Localization	T	265	53	174
Transcription	T	576	82	137
Gene expression	T	1738	356	722
Binding	T+	887	249	349
Regulation	T[,C]	961	173	292
Positive regulation	T[,C]	2847	618	987
Negative regulation	T[,C]	1062	196	379
TOTAL	-	8615	1795	3193

Table 4.1: Structure of the event types, their primary argument types and data statistics. Arguments abbreviate for (T)heme and (C)ause, with + marking arguments that can occur multiple times for an event and brackets defining optional arguments.

4.1.4 Corpus

The ST'09 corpus is divided in three distinct datasets: training data (800 articles), development data (150 articles) and the final test data (260 articles). These datasets all consist of PubMed abstracts extracted from the GENIA corpus (Kim *et al.*, 2008a). Stand-off annotation locating relevant GGPs is provided for all three datasets. Both training data and development data further include gold-standard annotations defining events as a specific trigger word in text, combined with one or several arguments. The event types and their statistics in the three datasets are shown in Table 4.1.

For the official participation in the challenge, participants had one week time to provide predictions for the final test dataset of 260 articles. Afterwards, new predictions on the test data could be evaluated using an online submission system maintained by the ST organizers, ensuring an objective evaluation of the methods. To avoid overfitting on the test set, all analyses in this study are performed on the development data unless explicitly stated otherwise.

4.2 Extraction framework

We have developed a novel ML framework for event extraction, building upon the previously introduced SVM classification pipeline for PPI extraction (Section 3.2). In this section, the design choices for this novel framework are described in detail.

The first crucial design choice for the classification framework involves its modularity, which either has a global or a local nature. A global approach implies inferring all plausible events in a single step and is highly computationally intensive. A local procedure on the other hand is characterized by a set of specific classifiers, creating predictions for distinct event types independently of each other. We have chosen a local

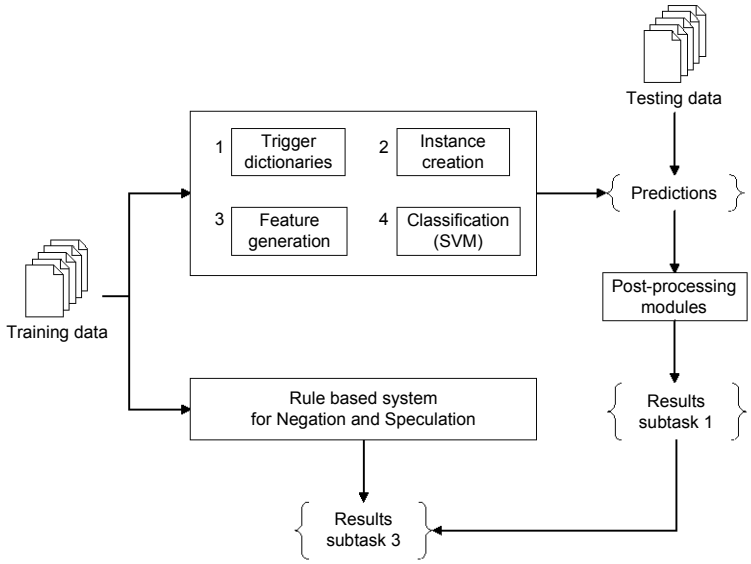


Figure 4.2: High-level overview of our event extraction framework. The box of four modules represents the generic pipeline that is first run in parallel for each physical event type and subsequently for the regulatory events.

approach by designing a generic pipeline which can be run in parallel for different event types. This pipeline consists of modules for trigger detection, instance creation, feature generation and classification. It can be employed for each event type for which sufficient training material is available.

The choice between a global and a local extraction method severely influences the size of the resulting datasets, limiting options for applicable classifiers. To illustrate, the global approach of Björne *et al.* (2009) yields a training set of 31,782 instances and 295,034 unique features. They state that the linear kernel of their multi-class SVM is the only practical choice for building a classifier with such large training sets. In contrast, the datasets obtained with our parallel design vary between 300 instances with 2000 features (protein catabolism) and 15,000 instances with 50,000 features (single-argument positive regulation). This reduced complexity enables us to experiment with more complex kernels such as a radial basis function. Performance results of various classifier setups are detailed in Section 4.2.5.

Our generic pipeline consists of 4 main modules: trigger detection through a dictionary approach (Section 4.2.2), instance creation (Section 4.2.3), feature generation (Section 4.2.4) and classification with an SVM (Section 4.2.5). This pipeline is run in parallel for each of the physical event types. The recursively defined regulatory events

Parser	p	r	F
Stanford	66.62	63.44	**65.00**
McClosky-Charniak	64.99	61.09	62.98
Bikel	60.94	57.87	59.37

Table 4.2: Performance of physical events for various parsers, benchmarked on the development data.

are subsequently predicted, repeating this step until no more events are found. In a final post-processing step, all event types and their corresponding predictions are merged into an integrated network. Such a network is also compiled from the training data and then used as a model to locate false-positive predictions and prune the corresponding edges. This process ensures consistency of the predictions made by the parallel classifiers (Section 4.2.6).

Figure 4.2 shows a high-level overview of the different modules in our framework. The core challenge of the ST'09, the extraction of biomolecular events from text, is referred to as subtask 1. The additional detection of negation and speculation is referred to as subtask 3. For this subtask, a rule-based algorithm was designed that achieved reliable results (Sections 4.2.7 and 4.2.8). We have not participated in subtask 2 which was concerned with the extraction of additional localization/site information such as the site of a protein phosphorylation.

4.2.1 Text preprocessing

To run an automated extraction algorithm, free text first has to be transformed into a machine readable format. To this end, data on sentence segmentation and tokenization has been made available by the ST organizers for all articles in the datasets. Furthermore, syntactic analyses created by various parsers is also provided with the dataset: both constituency trees and dependency graphs are available (Section 1.3.6).

A comparative study between various parsers providing both phrase structure parses and dependency graphs is shown in Table 4.2. This study included Bikel's implementation of Collins' parsing model (Bikel, 2004) and the Charniak-Johnson reranking parser using McClosky's self-trained model (McClosky *et al.*, 2006). In addition to these parses, made available by the ST organizers, we have employed the freely available Stanford parser (de Marneffe *et al.*, 2006). From these comparative experiments, we conclude that the Stanford parser performs best in our framework, yielding both higher precision and recall rates in comparison to the McClosky-Charniak parser and the Bikel parser.

Event type	Highest ranked trigger	Occurrence
Phosphorylation	'phosphoryl'	96%
Protein catabolism	'degrad'	76%
Gene expression	'express'	68%
Single-argument binding	'bind'	47%
Transcription	'transcript'	45%
Localization	'secret'	31%
Double-argument binding	'bind'	30%

Table 4.3: Most frequently occurring trigger in the training data, for each event type.

4.2.2 Trigger detection

The first step towards the extraction of biomolecular events from text concerns the challenge of trigger detection. A trigger is defined as a continuous interval of tokens and is linked to a certain event type, e.g. 'homodimerization' for a binding event. In the training data, triggers are annotated using their text offsets in a stand-off annotation format. A trigger word is not restricted to a particular set of part of speech tags, though verbs and nouns are the most commonly occurring cases. Furthermore, a trigger can consist of multiple consecutive words, e.g. 'binding partner'.

The challenge of trigger detection is tackled using carefully constructed dictionaries. First, all possible strings are collected by scanning the triggers in the training data and applying Porter's stemming algorithm (Porter, 1980). Our algorithm allows triggers to span multiple words, as this occurs frequently in the training data. This initial collection of all possible trigger strings results in entries of limited use, such as 'through' for binding and 'are' for localization. Such words lead to many negative and irrelevant instances as they are too general or too vague. To overcome this problem, the dictionaries were manually cleaned, only keeping specific triggers for each event type (e.g. 'interaction' for binding and 'secretion' for localization). Table 4.3 shows the single most occurring trigger for each physical event type in the training data.

During development, we noticed a significant difference between the triggers for single-argument binding events (e.g. 'homodimer', 'binding site') and those for binding events with multiple arguments (e.g. 'heterodimer', 'complex'). This motivated our choice to create two separate dictionaries.

With the same reasoning, regulatory events are also categorized into single-argument (one theme) and double-argument (an additional cause) events. Further analysing the nature of double-argument regulatory events, it became clear that a vast majority has a GGP specified as its causal argument. The dictionaries of these regulations are split up accordingly, differentiating between regulatory events caused by GGPs and those caused by other events. This automatically keeps the more general words (e.g. 'causes') out of the dictionaries of events regulated by GGPs (e.g. 'response').

4.2.3 Instance creation

The algorithm that defines instances in a ML framework has a severe influence on the balance of the datasets and ultimately on the performance of the framework. The nature of the event extraction task leads to unbalanced datasets, with much more negative examples than positive ones. This is due to the fact that GGPs may be involved in all possible event types, and each event type can be triggered by a plethora of possible nouns and verbs. As datasets with skewed class distributions are known to generate problems during classification (Monard and Batista, 2003), we try and filter out as many irrelevant negative instances as possible by introducing specific pre-processing methods and filters. This reduces unbalancedness of the datasets even prior to the classification step, ultimately limiting the number of false positives and improving precision of the predictions.

For the challenge of event extraction, an instance is defined as the combination of a trigger with one or more plausible arguments (Section 4.1.1). To locate suitable triggers in text, we implemented a fast algorithm using Radix trees[1], applying the dictionaries for each event type (Section 4.2.2). The careful construction of the dictionaries and the distinction between single-argument and double-argument bindings results in a significant drop in the number of binding instances in the training data, as irrelevant candidate instances such as a double-argument binding event with the trigger 'homodimer' are discarded. Consequently, the balancedness of the datasets is improved: from a total of 34,612 original binding instances (of which 2% positives) to 4,708 single-argument binding instances (11% positives) and 3,861 double-argument binding instances (5% positives).

For each identified trigger, candidate arguments are selected from the same sentence, as analysis on the training data reveals that 95% of all events are expressed within one single sentence. However, by defining an instance as any combination of a trigger co-occurring with its candidate arguments, too many irrelevant instances are generated, especially in long sentences. This results in imbalanced datasets with often less than 5% positives. For this reason, a negative-instances (NI) filter was implemented that applies some simple yet effective heuristics to reduce the dimensionality of the datasets, relying on the dependency parse and the length of the sub-sentence spanning the candidate event.

As described in Section 4.2.1, the Stanford parser was selected to create dependency graphs for each input sentence. For each instance, a minimal sub-graph is extracted, spanning its trigger and all arguments. Figure 4.3 shows the dependency parse of a sentence containing several biomolecular events. As an example, the sub-graph of the phosphorylation event spans nodes 17 and 19, while the positive regulation event triggered by 'essential' spans the sub-graph consisting of nodes 17, 19, 21, 23 and 24.

The NI filter enforces a cut-off on the size of the dependency sub-graph, as positive instances are known to be expressed in smaller sub-graphs than negative examples. Fig-

[1] Java implementation by Tahseen Ur Rehman, `http://code.google.com/p/radixtree/`

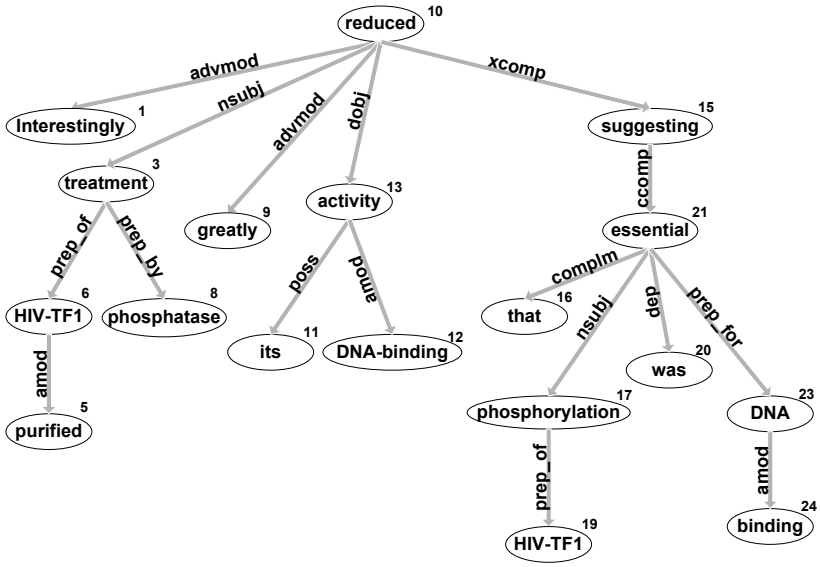


Figure 4.3: Example of a dependency graph for the sentence '*Interestingly, treatment of purified HIV-TF1 by phosphatase greatly reduced its DNA-binding activity, suggesting that phosphorylation of HIV-TF1 was essential for DNA binding.*', retrieved from PubMed article 1653950. Each word and punctuation mark in the original sentence is numbered, and these numbers are used as unique labels for the nodes. Note that not all words of the sentence appear as separate nodes in the graph.

ure 4.4 shows that the sub-graphs of double-argument binding instances are never larger than 10 edges, while artificially constructed negative instances may contain up to 18 edges. Only keeping instances with sub-graphs smaller than 8 edges would successfully discard 35% irrelevant negatives, while keeping 92% of the positive instances.

Similarly, the length of the sub-sentence spanned by a candidate event is used as a second parameter of the NI filter. Setting the threshold at 175 characters for double-argument binding events includes 99% of the positive examples, while removing about 20% irrelevant negatives. For training purposes, positive instances exceeding the NI filter cut-off are not actually removed as they add important information to the dataset.

The NI filter thus reduces the size and skewness of the datasets, resulting in faster classification pipelines. Furthermore, by identifying a large portion of negative instances even before they are processed by the classifier, a gain of 2.1 percentage points in F-score is obtained for physical events on the development set compared to a setup which does not use the NI filter.

For regulatory events, the NI filter becomes a true necessity as considering all can-

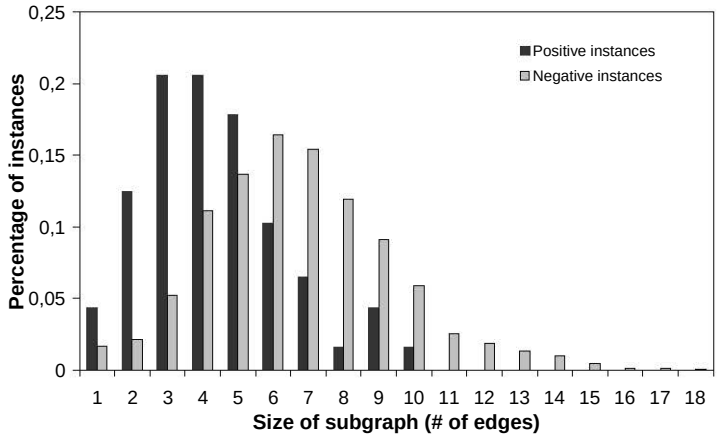


Figure 4.4: Distribution of double-argument binding instances, according to the size of the subgraph (training data).

didate argument sets from each sentence leads to datasets of tremendously high complexity. To illustrate, the filter reduces the number of negative training instances for single-argument positive regulations from 39,419 to 13,995, improving the percentage of positives from 4% to 10%, while reducing the feature set from 127,098 to 49,384 features.

The final distributions of positive and negative examples of physical events in the training data range between 7% and 51% (Table 4.4). It should be noted that the number of positive instances in Table 4.4 is lower than the actual number of positive examples in the training set, due to limitations of our instance definition method. However, a study regarding maximal recall shows that we do not remove too many true-positives (Section 4.3.1).

4.2.4 Feature generation

Our feature generation module is based on the corresponding algorithm for the PPI classification challenge (Section 3.2.2). PPIs were considered to be binary and symmetrical, and there was no specification of trigger words. Only one path in the dependency graph was analysed for each instance: the shortest path between the two candidate binding partners.

The present work on event extraction deals with much larger complex sub-graphs and thus requires a modified set of features to fully capture the semantics of each instance. As we are now dealing with graphs instead of trees, 'edge walks' are excluded (Section 2.3.2). Vertex walks are now the main source of information derived from the

Event type	neg.	pos.	
Phosphorylation	163	153	48%
Protein catabolism	175	96	35%
Gene expression	5,356	1,542	22%
Single-argument binding	3,548	522	13%
Double-argument binding	2,180	185	8%
Transcription	6,930	489	7%
Localization	3,415	249	7%

Table 4.4: Final distribution of instances for physical events in the training data.

dependency graph, consisting of two vertices and their connecting edge (Section 2.3.1). For the lexical variant of these patterns, GGP symbols and triggers are blinded, resulting in highly general features. The same principle is applied with theme and cause arguments for regulatory events. To illustrate using Figure 4.3, the nodes 17 and 19 are blinded as $causex$ and nodes 23 and 24 as $themex$ when processing the positive regulation event triggered by the word 'essential', blinded as $triggerx$. Resulting features for the vertex walks would then include '$triggerx$ nsubj $causex$' and '$triggerx$ prep_for $themex$'.

Blinding avoids overfitting of the classifier and simplifies the feature set (Section 2.4). However, in order not to lose valuable information, a few new features are introduced to enable reconstruction of the original information when necessary. The lexical tokens of the trigger (e.g. 'degradation') and the POS tags (e.g. 'noun') are stored as separate features. Similarly, additional features are included for regulatory events, marking whether the arguments are GGPs or events, and specifying the exact event type.

In addition to these features, each feature vector is augmented with lexical information. First of all, a bag-of-words approach is applied to all vertices in the sub-graph (Section 2.3.3), including highly informative words such as 'heterodimers'. This approach automatically excludes generic words such as prepositions, as they do not appear as individual nodes on the graph. Furthermore, trigrams derived from the sentence are added (Section 2.3.4). Trigrams are formed by combining three consecutive (stemmed) words in the sub-sentence delimited by the trigger and GGP offsets in the text. They capture common phrases such as 'high levels of'.

The size of the sub-graph and the length of the sub-sentence are also included in the feature vector. Even though these two values are used in the previous step as parameters for the NI filter, they are still relevant for classification. Indeed, instances that only just pass the filter still have a higher chance of being negative.

In the training phase, the processing of each instance generates different patterns, and each pattern is stored as a numeric feature in the feature vector (Section 2.5). During testing, the algorithm counts how many times each feature is found for each instance. This results in very sparse and high-dimensional datasets.

Event type	Features
Protein catabolism	1,883
Phosphorylation	2,185
Double-theme binding	11,228
Localization	18,121
Single-argument binding	21,332
Transcription	30,306
Gene expression	31,332

Table 4.5: Dimensionality of the datasets.

Table 4.5 shows the dimensionality of the datasets for all event types. Protein catabolism has the lowest dimensionality with 1,883 features, while transcription and gene expression produce over 30,000 features.

4.2.5 Classification

Our event extraction framework needs a classifier able to deal with thousands of instances, thousands of features, and a class imbalance of up to 93% negative instances. To this end, the state-of-the-art LibSVM (Chang and Lin, 2001) classifier is used, as implemented by WEKA (Hall *et al.*, 2009). Analyses were conducted experimenting with both the linear kernel and the radial basis function (RBF), and various strategies for parameter tuning were implemented.

The linear kernel requires tuning of the cost parameter c, which was implemented with an internal 5-fold CV loop performing a grid search on the training portion of the data. Values between 2^{-5} and 2^{14} were tested, and the one producing the best F-score was automatically selected for each dataset.

However, a more complex problem arises when both the parameters γ and c of the RBF kernel have to be tuned simultaneously. Possible strategies include a straightforward grid search or a more advanced pattern search. A pattern search is a parameter optimization algorithm that starts at the center of the search range and subsequently explores small steps in each direction for each parameter (Lewis and Torczon, 1999). If the fit of the model improves, the search center moves to the new point and the algorithm is repeated. If no improvement is found, the step size is reduced and the search executed again. The pattern search stops when the search step size is reduced to a specified minimum value. Even though this technique is computationally less expensive than a full grid search, there is a danger of ending in a local optimum. For this reason, a combined search strategy was designed, which first employs a grid search to determine rough values for γ and c. These values are then fine-tuned by the pattern search. This strategy should avoid local optima, while at the same time being less computationally expensive than a full grid search.

Kernel	parameters	p	r	F
RBF	tune c	66.62	63.44	**65.00**
RBF	tune c and γ	30.00	28.75	29.36
Linear	tune c	64.58	61.34	62.92

Table 4.6: Performance of physical events for various classification settings.

Despite careful design of the tuning algorithm, performance on the development test set drops dramatically when using the RBF kernels tuned for both parameters (Table 4.6). However, comparing this classifier to one that only has been tuned for the parameter c, about equal performance is reached within the internal CV loop. The most plausible explanation for the final drop in performance thus seems to be that the complex search strategy severely overfits the parameters on the training portion of the data. Eventually, γ was therefore kept at its default value.

While the linear kernel runs faster, it performs slightly worse than the RBF kernel, with a drop in performance of about 2 percentage points in F-score. Consequently, the RBF kernel was chosen for all other analyses.

Finally, we tested the influence of assigning higher weights to positive training instances, to try and correct for the imbalanced nature of the data, but this had almost no effect on overall classification performance.

4.2.6 Post-processing

LibSVM produces numeric values between 0 and 1, requiring a cut-off to define the separation between true and false instances. This problem was tackled by chosing the best SVM cut-offs on the development data for each classifier. However, an additional post-processing step is necessary to obtain a final set of coherent predictions.

First, two triggers of different event types might overlap, based on the same words in the text. For example, the trigger 'expression' can lead to both a transcription and a gene expression event, but not at the same time[2]. In such a case, only the prediction with the highest SVM score is selected. However, thanks to a careful construction of the dictionaries (Section 4.2.2), their mutual overlap is rather small, and this post-processing module thus has almost no influence on performance.

Further, one trigger might be involved in different events from the same event type. For example, the sentence 'it induces expression of *STAT5*-regulated genes in *CTLL-2*, i.e. *beta-casein*, and *oncostatin M* (*OSM*)' mentions two gene expression events based on the trigger 'expression', one involving *beta-casein*, and one involving *OSM*. For these two events, the sub-graphs will be very similar, resulting in similar features and SVM scores. However, often a trigger only leads to one true event, while all other candidates

[2]This is a property of the ST corpus. Even though transcription can be defined as a sub-class of gene expression, a single best event type is chosen for each textual occurrence.

from the same event type are false positives. We have carefully benchmarked this hypothesis, and found that for protein catabolism and phosphorylation, better performance could be achieved by only keeping the top-ranked prediction. Up to 5% in F-score could be gained for these events. This is due to the fact that for these two event types, usually only one true event is linked to each trigger.

4.2.7 Negation

As an optional subtask of the Shared Task 2009, statements that are negated or speculated had to be recognised. For event negation, there are three major categories of statements:

1. A negation construct in the close vicinity of the trigger (e.g. 'no', 'failure to').
2. A trigger already expresses negation by itself (e.g. 'non-expressing', 'immobilization').
3. A trigger in a certain sentence expresses both positive as negative events. In this case, the 'but not' pattern is often used (e.g. 'overexpression of Vav, but not SLP-76, augments CD28-induced IL-2 promoter activity').

We have created a custom-made rule-based system to process these three categories. The rules make use of small dictionaries collected from the training data. For rule 1, we checked whether a negation word appears right in front of the trigger. Rule 2 uses a list of inherent negative triggers deduced from the training set. Rule 3 finally looks for certain patterns such as 'but not' or 'whereas', negating only the event involving the GGP mentioned right after that pattern.

4.2.8 Speculation

There are two major reasons why the description of an event should be viewed as speculative. These categories are:

1. Research hypothesis: the authors state the topic of their study, without knowing the true results (yet). This is often indicated with expressions such as 'we have examined whether (...)'.
2. Data interpretation: the authors formulate an interpretation and conclusion to explain the results of a certain experiment. Specific speculation words such as 'might' or 'appear to' often occur right before the trigger.

Similarly to detecting negation, a list of relevant expressions was compiled from the training data and used to implement a simple rule-based system. Rule 1 was implemented by checking the presence of such an expression in a range of 60 characters before the trigger and up to 60 characters after the trigger. Rule 2 is applied within a smaller search window: only 20 characters right before the trigger.

	Event type	Max. recall
Physical events	Protein catabolism	100.00
	Phosphorylation	95.74
	Gene expression	91.57
	Transcription	90.24
	Localization	84.91
	Binding	78.23
Regulatory events	Regulation	46.15
	Negative regulation	43.88
	Positive regulation	39.71
Modifications	Negation	28.97
	Speculation	25.26

Table 4.7: Maximal recall for the development data, in percentages.

4.3 Results

In the first part of this chapter, we have described the machine learning framework developed for event extraction as defined by the ST'09. In the following sections, we first detail the results of our system in the official participation in this Shared Task. Due to the time constraints of this challenge, we were not able to optimize the design of our framework. In the months following the official task, we have further extended and improved upon the system, ultimately resulting in a relative performance gain of 10%. These improvements, as well as a few novel analyses, are described in detail. Finally, we present a feature selection study using ensemble feature selection, obtaining more cost-effective classifiers while at the same time gaining valuable insight into the event classification task.

4.3.1 Benchmarking on the development data

Physical events

To evaluate maximal recall of our instance extraction method, we executed an evaluation using an all-true classifier. As can be seen in Table 4.7, maximal recall is quite high for almost all physical events. This proves good coverage of the dictionaries (Section 4.2.2) and shows that the NI filter does not remove too many correct instances (Section 4.2.3). The relative drop in performance for binding events could be due to the fact that these are often expressed across sentences, which would not be found by our method.

Results of the final performance are detailed in Table 4.8. For most events, very high precision is achieved, binding again being a notable exception.

Inspecting the F-measures, transcription, gene expression and phosphorylation all perform between 67 and 77%, while localization and protein catabolism have an F-score

Event type	r	p	F
Protein catabolism	80.95	89.47	85.00
Localization	77.36	91.11	83.67
Phosphorylation	68.09	88.89	77.11
Gene expression	70.79	79.94	75.08
Transcription	60.98	75.76	67.57
Binding	45.16	37.21	40.80
Total (physical)	62.45	64.40	63.41
Negative regulation	30.10	41.26	34.81
Regulation	23.67	41.67	30.19
Positive regulation	21.56	38.00	27.51
Total (regulatory)	23.63	39.39	29.54
Task 1	41.03	53.50	**46.44**
Negation	15.89	45.95	23.61
Speculation	20.00	26.87	22.93
Total (neg/spec)	17.82	33.65	23.30
Task 3	38.77	52.24	**44.51**

Table 4.8: Performance of all events for the development data, measured with the original system.

of more than 83%. It becomes clear that binding is the most difficult event type, with a performance of 40.80% F-score. Unfortunately, this event type covers 44% of all physical events, greatly influencing total performance. Average performance of predicting physical events results in 63.41% F-score.

Regulatory events

The performance of regulatory events greatly relies on the ability of our system to accurately predict physical events. Indeed, one FN physical event can lead to multiple FN regulatory events, and the same holds for FPs. Furthermore, events across sentences are not extracted, leading to even more FNs. Finally, regulatory events are expressed with a much wider variety of interaction words. As a result, some instances in the test data can not be extracted because the triggers are missing from the dictionaries.

To study maximal recall of the regulatory events, we have again applied an all-true classifier. Table 4.7 shows that the recall of the regulatory events is never higher than 50%.

As regulatory events can be the arguments of other regulatory events, the regulation pipeline should be run repeatedly until no more new events are found. In our experiments, we have found that even the second recursive run did not lead to much better performance, and only a few more regulatory events were found.

Final results are shown in Table 4.8. With recall being rather low, between 21% and 30%, relatively good precision is still achieved: around 40% for each of the three

Team	Country	Task 1	Task 2	Task 3
UTurku	Finland	**51.95**	-	-
JULIELab	Germany	46.66	-	-
ConcordU	Canada	44.62	-	**42.52**
UT+DBCLS	Japan	44.35	**43.12**	-
VIBGhent	Belgium	40.54	-	37.80
UTokyo	Japan	36.88	-	-
UNSW	Australia	34.92	-	-
UZurich	Switzerland	34.78	-	-
ASU+HU+BU	US, Hungary, Germany	32.09	29.26	29.57
Cam	UK	30.80	-	-
UAntwerp	Belgium	30.58	29.27	-
UNIMAN	UK	30.35	-	-

Table 4.9: Official performances (F-score) for the BioNLP Shared Task 2009 (Kim *et al.*, 2011), showing the top 12 out of 24 participants (sorted by task 1). The table further includes the results of the 3 best performing systems of task 2 and 3.

regulation types. On average, the F-score is almost 31% for the regulatory events, which is significantly lower than the performance of physical events. On average, an F-score of 46.44% is obtained on the development data for task 1.

Negation and speculation

The performance of this subtask depends heavily on the performance of subtask 1. Again we have applied an all-true classifier to determine maximal recall (Table 4.7). Less than 30% of the events necessary for task 3 can be found with our setup; all of these FNs are due to FNs in task 1.

Table 4.8 shows the final results: around 23% F-score is achieved on the development data. We take into consideration that according to the maximal recall study, only 29% of the necessary events for negation were extracted by task 1. In the final results, 16% of all the negation events were found. This means that our rule-based method by itself achieves about 55% recall for negation. Similarly, the system has a recall of 80% for speculation when only considering events found in task 1. The simple rule-based system thus performs reasonably well.

4.3.2 Scoring and ranking on the final test set

In the official BioNLP Shared Task of 2009, our system obtains a 5^{th} place out of 24 participating teams with an F-score of 40.54% for subtask 1 (Table 4.9). For subtask 3 of finding negation and speculation, a second place is obtained with 37.80% F-score.

The official results for each of the event types are shown in Table 4.10. As on the

Event type	r	p	F
Phosphorylation	56.30	89.41	69.09
Gene expression	59.42	81.56	68.75
Protein catabolism	64.29	60.00	62.07
Localization	43.68	78.35	56.09
Transcription	39.42	60.67	47.79
Binding	38.04	38.60	38.32
Total (physical)	50.75	67.24	57.85
Negative regulation	22.96	35.22	27.80
Positive regulation	17.19	32.19	22.41
Regulation	10.65	22.79	14.52
Total (regulatory)	17.36	31.61	22.41
Task 1	33.41	51.55	**40.54**
Negation	10.57	45.10	17.13
Speculation	8.65	15.79	11.18
Total (neg/spec)	9.66	24.85	13.91
Task 3	30.55	49.57	**37.80**

Table 4.10: Official performance for the BioNLP Shared Task 2009, broken down by event type.

development data, the binding event type performs worst among all physical events, and the same trend is found when analysing results of other teams. However, in general high precision results are achieved: 67% for physical events, 52% on average on subtask 1, and 50% on average on subtask 3. Another trend reported also by other teams, is the relatively high performance of physical events, compared to the prediction of regulatory events.

The three types of physical events that perform best in our ML framework correspond to the datasets with the highest percentage of positive examples: phosphorylation, gene expression and protein catabolism (Table 4.4). Furthermore, each of these 3 types is linked to a possible trigger that accounts for more than 65% of the training examples (Table 4.3). In contrast, binding events are much more difficult to identify as they are expressed with a broad spectrum of possible triggers. This is also illustrated by the maximal recall study, as new triggers can not be identified in the test set if they have not occurred in the training data.

Comparing these official results to those obtained on the development data (Table 4.8), there is a performance drop for the physical events of about 6 percentage points. This loss is propagated to the regulatory events and to negation and speculation, each also performing about 6 percentage points worse than on the development data. We believe this drop in performance might be due to overfitting of the system during training. The SVM cut-offs have been tuned on the development data, but they might not be ideal for the final test set.

4.3.3 System improvement

As detailed in the previous section, our official result for the Shared Task reports a global performance of 40.54% F-score on the test data, achieving 5^{th} place out of 24 participants (Van Landeghem *et al.*, 2009). In subsequent work, few improvements were made to the original system (Van Landeghem *et al.*, 2011c).

First, the construction of the trigger dictionaries was improved upon (Section 4.2.2). Originally, the trigger dictionaries were cleaned manually, only keeping specific triggers for each event type (e.g. 'interaction' for binding and 'secretion' for localization). Inspired by the work of Buyko *et al.* (2009), we now use their proposed formula to calculate the importance of an event trigger t_i for a particular event type T:

$$Imp(t_i^T) = \frac{f(t_i^T)}{\sum_{p=0}^{n} f(t_p^T)}$$

where $f(t_i^T)$ is the frequency of the event trigger t_i of event type T in a training corpus ($i = 0, \ldots, n$). By applying a cut-off value of 0.005, only those words are kept that are informative enough for a specific event type. In contrast to the work of Buyko *et al.* (2009), this measure is not used for event trigger disambiguation as words are allowed to be included in trigger dictionaries of different event types. This choice was motivated by analyses of the training data, which has shown that the same word in text may actually trigger multiple events of different types. The word 'overexpression' is a frequently recurring example, and is often linked to both a gene expression event as well as a positive regulation event. Following the same reasoning, we removed the post-processing step which would resolve overlapping triggers of different event types to the most likely one (Section 4.2.6). While most other systems disambiguate the type of a trigger in an early stage of the pipeline, our parallel approach can easily avoid this issue and thus model the data more truthfully.

A few more improvements to the post-processing module (Section 4.2.6) were implemented. Originally, cut-off values on the LibSVM scores were determined by evaluating the classifiers on the development data. However, this methodology had a serious drawback: new cut-off values had to be defined whenever a new classifier was trained or when other parameters had been changed in the system. We now approach the problem of selecting the right predictions from a different angle, ensuring global consistency of the final set of predictions rather than making local decisions. As a first step, all instances from the testing set are collected and merged into an integrated network with weighted edges according to their LibSVM scores. Global consistency of the network is then imposed by using a model obtained from the training data. As instances are created in the same way for both training and testing datasets, the percentage of positives in the training set provides a reasonable estimate for the number of positives in the testing set. Furthermore, this measure is independent of the classifier. By keeping the top ranked predictions until a certain percentage of positives is reached, a gain of 1.6 percentage points in F-score for physical events is obtained on the development data.

Event type	r	p	F
Phosphorylation	77.04	70.27	73.50
Gene expression	62.74	82.21	71.17
Protein catabolism	64.29	50.00	65.25
Localization	43.10	80.65	56.18
Transcription	57.66	53.02	55.24
Binding	33.43	42.03	37.24
Total (physical)	54.68	67.69	60.49
Negative regulation	22.43	46.20	30.20
Positive regulation	22.99	37.79	28.59
Regulation	15.12	28.21	19.69
Total (regulatory)	21.48	37.85	27.40
Task 1	37.43	54.81	**44.48**

Table 4.11: Final performance on the test data, as obtained with the improved framework.

This method creates an additional advantage, allowing for the development set to be used as training data, as the cut-off values are now obtained from the training data only. The classifiers can thus be trained on the merged articles from both training and development data, extending the training set from 800 articles to 950. Obviously, this setup is only used to produce results on the final test set.

To fine-tune the model and ensure global consistency of the predictions even further, more detailed information is extracted from the training data. For example, there are no training examples of two regulatory events with the same arguments but switched semantic roles. Consequently, when these kind of network 'loops' happen in the test data, the edge with the highest weight is selected.

Finally, a few minor improvements were made to the instance creation module, utilizing 'Equivalence' relations to create more positive examples. As part of the manually annotated datasets, these relations mark synonyms and acronyms referring to the same biological entity. For example, in the sentence '*The c-Rel homodimer has a high affinity for interleukin-6 (IL-6)*', *interleukin-6* and *IL-6* are annotated as equivalent entities. In the gold-standard event annotation, only *c-Rel* and *IL-6* are annotated as binding partners. A new function was thus implemented to additionally recognise the binding event of *c-Rel* and *interleukin-6*.

The new and improved system achieves 37.43% recall, 54.81% precision and 44.48% F-score on the final test set (Table 4.11). During the development of these system improvements, only a few experiments were run on the final test set, in order not to overfit the classifiers to the test data. All other analyses were performed on the development dataset. Remarkably enough, a higher relative gain in performance was obtained on the final test data (10%) than on the development data (5%). This indicates that the original system could have been slightly overfitted to the development data, while the improved system is more general and can better cope with new data.

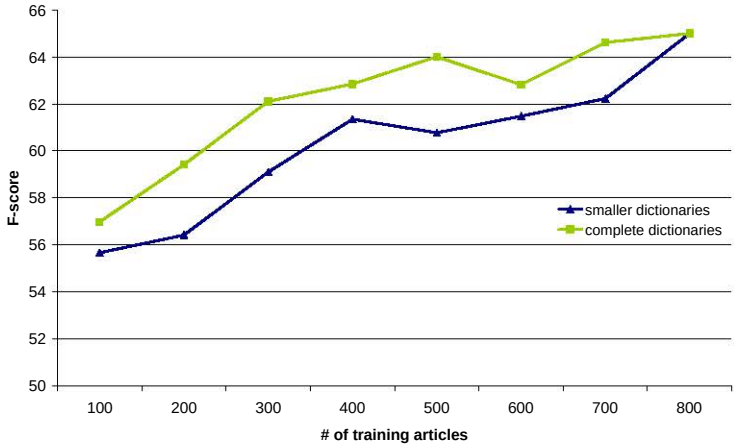


Figure 4.5: Learning curve of physical events, benchmarked on the development data.

4.3.4 Learning curve

The organization of the Shared Task 2009 greatly popularized the event extraction challenge, resulting in the development of many novel extraction frameworks. However, state-of-the-art systems are still not able to predict events with higher performance than 60%. As the best performing systems are based on machine learning, we want to investigate whether the training data is sufficiently large to cover all possible patterns in the test data. To assess the influence of the size of the training data on the final performance, the training data was divided into 8 portions of 100 articles. The first classifier was trained using one portion only. By incrementally adding 100 articles to train the next classifier, prediction performance started rising (Figure 4.5).

Two versions of this experiment were conducted, experimenting with different dictionaries for the extraction of triggers. The first one was run with dictionaries constructed from all 800 training articles ('complete dictionaries'). For the second version however, the dictionaries were only based on the data available in the smaller portion of the training data ('smaller dictionaries'). The discrepancy between these two versions clearly shows the added value of having better curated dictionaries. As both curves do not indicate a level of saturation quite yet, we hypothesize that supervised learning systems might benefit from even more training data.

4.3.5 Precision vs. recall

In most retrieval systems, an inverse relationship exists between recall and precision (Section 1.3.8). An important advantage of a ML framework lies in its ability to be

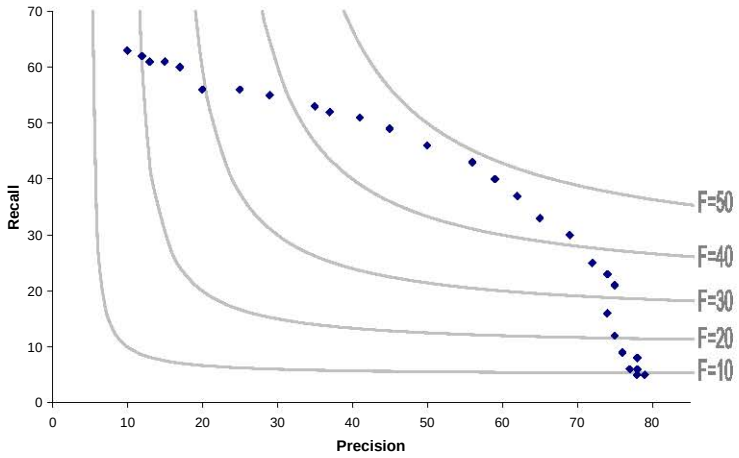


Figure 4.6: Precision-recall curve for predicting all events: dots indicate the performance when varying the LibSVM thresholds. Grey lines each mark a constant F-score level.

tuned to achieve either good precision or good recall. The nature of an application often prefers one of the two. High-precision systems may be preferred for providing reliable textual evidence for claims made in structured databases. On the other hand, biological experimentalists might especially be interested in low-confidence predictions as these may represent interesting hypotheses for new studies. Humans are particularly good at selecting the right information from a collection of noisy predictions, as illustrated daily by the success of Google[3], a search engine that often produces thousands of informative hits of which only a few are truly relevant to the user.

To accommodate for specific needs regarding either high precision or high recall, the parameters of the post-processing module (Section 4.2.6) can easily be tuned to create a new set of predictions accordingly. Instead of selecting about the same percentage of positives as found in the training data, this number can either be reduced or increased, trading recall for precision and vice versa. To illustrate, a precision-recall curve was compiled for the prediction of all events in the development data (Figure 4.6). The highest F-score (48.62%) is achieved at 55.67% precision and 43.15% recall, corresponding to an exact mapping of percentages between the training and testing set (Table 4.4). If only the top 40% of the original predictions is selected, a precision of 74.90% is achieved, recall of 21.24% and F-score of 33.10%. The highest possible precision rate is 79.17%, but performance then drops dramatically to 9.42% F-score.

As benchmarking on the final test set is limited, a similar graph for the final system trained on 950 articles was not produced, but we expect the results to be similar. To

[3]Google Inc, `http://www.google.com/`

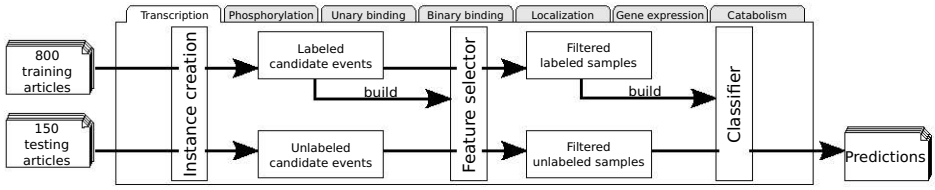


Figure 4.7: Overview of the general event extraction pipeline. For each event type, candidate events in the training data are used to create a feature selector, which is subsequently applied for feature selection of both training and testing instances. Finally, a classifier is built with the filtered training samples and applied for predicting events in the test set.

illustrate, one additional experiment was run on the final test data using the 40% factor, achieving 73.77% precision, 17.85% recall and 28.74% F-score. These numbers outperform the system of manually written rules that achieved the highest precision among all participants in the official Shared Task with 71.81% precision, 13.45% recall and 22.66% F-score (Cohen *et al.*, 2009). These results show that a ML framework can compete with a rule-based method even when high precision is required.

4.4 Ensemble feature selection

The BioNLP ST provides the community with standardized evaluation measures, enabling a meaningful comparison between various systems. Analysis of the official results of the 24 participants has indicated that ML systems using SVMs dominate the top ranked systems (Kim *et al.*, 2009). The most popular approach, using carefully designed rules, generally provides higher precision. However, SVMs can also be tuned to achieve such high levels of precision, while maintaining high overall performance (Section 4.3.5). Consequently, SVMs are gaining popularity in the BioNLP community.

Even though ML algorithms have been shown to achieve excellent performance, their typical characteristic of being a 'black box' often prohibits the end-user to fully understand the nature of the predictions. This is definitely the case for event extraction from text, as typical datasets contain thousands of instances and thousands of features. Feature selection can help to gain more insight into this data abundance, by identifying features that are highly discriminative and marking these for the end-user. At the same time, this insight can be applied to develop more accurate NLP tools.

We present the first application of a robust ensemble FS method for creating more cost-effective classifiers for event extraction from text (Van Landeghem *et al.*, 2010). This algorithm only keeps the most informative features, drastically reducing the dimensionality of the feature vectors and thus the complexity of the classification algorithm. Figure 4.7 presents a schematic overview of the feature selection pipeline.

The next sections present the main results of this study. First, we discuss the results for feature selection stability (Section 4.4.2) and describe the classification performance of the enhanced framework (Section 4.4.3). Further, Section 4.4.4 discusses the relative importance of the various feature types and Section 4.4.5 offers many in-depth analyses of the discriminative power of individual features. As the prediction of regulatory events greatly depends on the ability to predict physical events (Section 4.3.1), most experiments were only performed for the physical events. Conclusions drawn from this setup can easily be extended to the entire framework.

4.4.1 Methodology

We used the recently introduced concept of ensemble feature selection (Saeys *et al.*, 2008). This approach builds on the idea of ensemble classification by using multiple weak feature selectors to build a single robust one. These weak feature selectors are created by bootstrapping the training data and then building a support vector machine. The weights of the support vectors determine the rank of the features, and individual rankings are aggregated in a consensus ranking using linear aggregation (Abeel *et al.*, 2010). Bootstrapping is done as sampling with replacement to obtain a bootstrap of the same size as the training set. Training sets for the individual runs are created by sampling without replacement 90% of the entire training set.

Stability of feature rankings is measured using the Kuncheva consistency index (Kuncheva, 2007):

$$\mathrm{KI}(\mathbf{f}_i, \mathbf{f}_j) = \frac{r \cdot N - s^2}{s \cdot (N - s)}$$

where $\mathbf{f}_i$ and $\mathbf{f}_j$ are the top features of ensemble ranking i and j, $s = |\mathbf{f}_i| = |\mathbf{f}_j|$ denotes the signature size, $r = |\mathbf{f}_i \cap \mathbf{f}_j|$ equals the number of common elements in both signatures and N represents the original number of features. A higher Kuncheva index indicates a larger number of commonly selected features in both signatures.

The signature size can either be expressed as the total number of retained features, or as the percentage of the feature space that is retained after feature selection. For knowledge discovery, the signature size typically needs to be small enough to support manual analysis. For classification however, classification performance and feature reduction have to be optimized jointly.

4.4.2 Stable feature selection

Figure 4.8 plots the distribution of the feature selection stability in function of the number of bootstraps used for the consensus ranking. From this figure, it is clear that using more bootstraps to create the consensus ranking has a beneficial effect on the stability of the selected features. Even though there are still small gains, the stability improvements saturates around 60 bootstraps. While the figure is generated from the dataset on

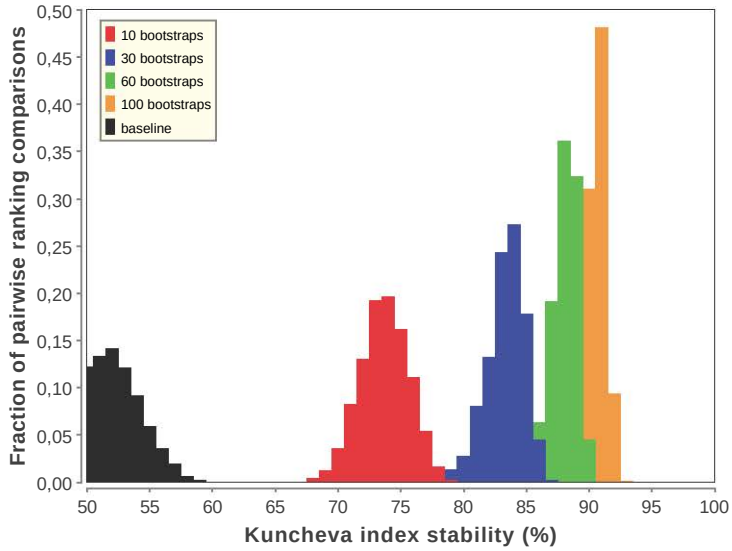


Figure 4.8: Feature selection stability improvements by using more bootstraps for the single-argument binding event. Distributions are plotted for 10, 30, 60 and 100 bootstraps and the baseline feature selection when retaining 25% of the features. The stability is measured with the Kuncheva index between all pairwise combinations of consensus rankings.

single-argument binding, similar graphs are obtained for the other event types (data not shown). The increase in stability from baseline to a 100 bootstrap consensus ranking is between 20% (on the transcription set) and 43% (on the protein catabolism set).

More stable feature selection means less variation of the selected features, which has two main benefits. First of all, stable feature selection identifies more meaningful features and allows the construction of better performing classifiers (Section 4.4.3). Furthermore, it enhances the interpretability of the selected features (Sections 4.4.4 and 4.4.5).

4.4.3 Enhanced accuracy and reduced dimensionality

When irrelevant features can be eliminated from the dataset, an SVM should have an easier task distinguishing true predictions from false ones, resulting not only in faster classifiers but also in enhanced performance. To test this hypothesis, we evaluated the performance of the classifier when using only a small fraction of the original feature space. We compare these results with the global baseline performance of our system (65.02% F-score), which is the performance of physical events as obtained by the improved system (Section 4.3.3) on the development set.

Signature	Min.	Max.	Avg.
75%	64.85	65.33	65.26
50%	65.60	66.43	65.88
30%	64.94	66.60	65.86
25%	65.51	66.82	**66.14**
20%	65.08	66.56	65.85
10%	61.75	64.90	63.59

Table 4.12: Classification performance for all 100 FS runs, showing minimum, maximum and average F-score for global event extraction. The initial baseline without feature selection is 65.02 F-score.

Table 4.12 presents the classification results when incorporating feature selection. Evaluation is performed on 100 distinct FS runs, and the table reports on minimum, maximum and average performance across these runs. The results clearly show that feature selection improves the classification performance: the combined model consistently outperforms the baseline performance at signature sizes of 20% and more. Further experiments indicated that performance peaks around 25% of the feature space with minimal variance between the folds. Performance starts dropping below baseline with a signature size of about 10%.

These results prove that our feature selection algorithm successfully discards irrelevant features, producing a dimensionality reduction of 75% and average classification improvement of 1.12 percentage points in F-score. As this result validates the output of the FS algorithm, it also creates the opportunity to analyse the top-ranked features in more detail. By analysing which features are highly important to the support vector machine, we will be able to gain some insight into this black-box algorithm. This is not only beneficial for the end-user, providing clues why events are predicted, but will also be applicable for enhancing the implementation of machine learning frameworks for event extraction. Both applications are discussed in the next two sections.

4.4.4 Relative importance of feature types

Section 4.2.4 discussed the various classes of feature types used in our framework. To assess the relative importance of each type, we have analysed the consensus ranking produced by aggregating the results of the 100 FS runs. Figure 4.9 details the results for the dataset on single-argument binding. Highly similar graphs were obtained for the other datasets and overall conclusions follow the same trend.

Figure 4.9 depicts the relative rate at which each of the feature types is being selected at each step of the FS algorithm. This analysis shows that the features expressing syntactic information about the trigger words are overrepresented in the top ranked features, i.e. they are being selected first. About 90% of all syntactic triggers are present in the top 5% of the consensus ranking and all of them are present within the top 20% features.

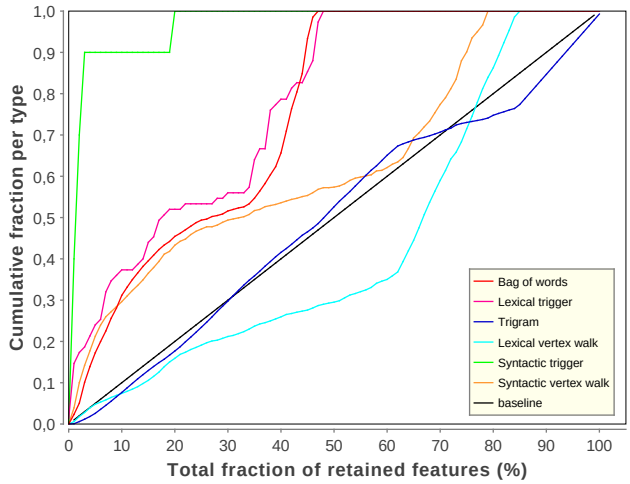


Figure 4.9: Feature selection order for the dataset on single-argument binding. The x-axis shows the total fraction of selected features in the feature set, while the y-axis displays the fraction of features of one specific feature type. The black line indicates a random feature selection baseline method.

Even when selecting less than 50% of the total feature space, all lexical information about triggers as well as all BOW features are already included. Consequently, these feature types appear to be highly relevant and include practically no irrelevant features.

Vertex walks express grammatical relations between the words of the dependency graphs. The features of the syntactic variant are highly overrepresented in the top 20% of the ranking, but their relative increase diminishes afterwards. The lexical counterpart appears to be much less informative in general.

Finally, trigrams resemble the baseline in the top 70% of the features, and form the entire last 20% of the ranking. From these results, we can conclude that the feature generation method produces many irrelevant trigrams. We have analysed these bottom-ranked trigrams and found that many originated from 3 consecutive words in different phrases, such as ‘subunits and the’. Here, the conjunctive ‘and’ links two distinct noun phrases. It could thus be more beneficial to extract trigrams only from within the same noun or verb phrase (e.g. ‘interacts directly with’).

Considering the striking findings concerning trigram features, we have condicted new experiments, each time excluding one specific type of feature (Table 4.13). Clearly, even the trigrams contribute to the global performance, as there is a small drop in F-score when leaving them out.

FS	Features	p	r	F
Manual	All	66.62	63.44	65.02
Manual	All except BOW	61.34	64.07	62.68
Manual	All except trigrams	62.21	65.15	63.64

Table 4.13: Performance of physical events for different feature sets, tested with the RBF kernel.

4.4.5 Individually discriminating features

To gain even deeper insight into the most discriminating features, we have analysed the feature ranking for each distinct event type across all 100 folds. For each ranking, the top 100 features were taken into account. Even though this top 100 is too small to capture the complexity of event extraction in a classification setting, analysis of the most frequently occurring features in the top 100 provides strong clues on the properties of the most discriminating features and allows us to learn interesting aspects of the feature generation process.

Each individual feature appearing at least once in the top 100 is assigned a score, by counting the number of times it occurs in a top 100 and assigning higher weights to higher ranked features. Subsequently, tag clouds of these features were generated by scaling their font size according to their weight and applying a color coding scheme that shows whether the feature mainly occurs in negative samples (bright red), in positive samples (blue), or equally in both (purple). To correct for the large class-imbalance present in most datasets, the actual rate was normalized using the expected rate in each dataset, by taking into account the specific class distribution (Table 4.4).

In this section, we discuss some of the most interesting tag clouds in detail. The chosen tag clouds represent various event types as well as various feature types, and the words appearing in them are transformed to their stemmed and lowercase variants.

Figure 4.10 shows the most informative trigger words for the localization dataset, identifying crucial words such as 'local(ization)' and 'secret(ion)' as highly relevant trigger words for this dataset. However, at the same time we notice that 'express' and 'presenc/t' also rank high, but indicate negative events. Consequently, these trigger words should probably have been eliminated from the dictionaries in the first place. Indeed, the formula for $Imp(t_i^T)$ does not take into account the balance between positive and negative examples for a certain trigger (Section 4.2.1). It would thus be beneficial to incorporate this information into the formula, eliminating negative candidate events even before classification, while at the same time reducing the dimensionality of the datasets. However, this is a complex problem as the frequency of trigger words is likely to be different in the training and testing data.

There is another lesson to be learned from Figure 4.10: the two stemmed words 'presenc' and 'present' are treated as distinct triggers, even though they refer to a similar concept. This finding indicates an important shortcoming of stemming, which ap-

Figure 4.10: The most discriminative lexical triggers for localization events.

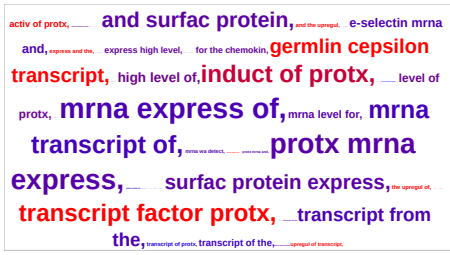


Figure 4.11: The most discriminative trigrams for transcription events.

plies simple suffix-stripping rules but does not map similar concepts to the same word. Lemmatization, in combination with a dictionary lookup to identify related words, could solve this problem and provide even better generalization of the feature vectors.

As a next example, Figure 4.11 presents the most informative trigrams for the transcription dataset. The pattern 'transcript factor $protx$' strongly hints towards a negative example, as it indicates that the text defines the protein as a particular transcription factor rather than actually discussing transcription of that protein.

In contrast, the framework has found several interesting positive patterns involving mRNA expression. 'mRNA' is also selected as the most informative BOW-feature for transcription (data not shown). This clearly shows that our framework is capable of deducing relevant biological knowledge from the training data, without having to turn to external databases, ontologies or expert knowledge. This characteristic is very valuable, as an ideal text mining framework does not rely on any external information, but can instead process information not yet indexed by external resources.

The tag cloud for trigrams in the phosphorylation dataset shows similar examples involving 'i kappa b alpha' (Figure 4.12), while at the same time indicating a limitation of the feature representation: patterns of more than 3 words can not be efficiently captured. While the various parts are present ('i kappa b' and 'kappa b alpha'), it could be valuable to create additional features considering N-grams with $N > 3$ in a new version of the text mining algorithm.

An additional problem is caused by the heterogeneity of word usage by various authors, an intrinsic property of natural language. Indeed, in some of the text, 'i kappa b alpha' is referred to as 'iKappaBAlpha', 'IkappaB-alpha' or 'I kappa B-alpha'. Our current feature vectors are incapable of linking these terms to the same concept. Here, a

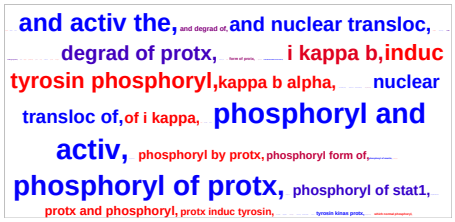


Figure 4.12: The most discriminative trigrams for phosphorylation events.

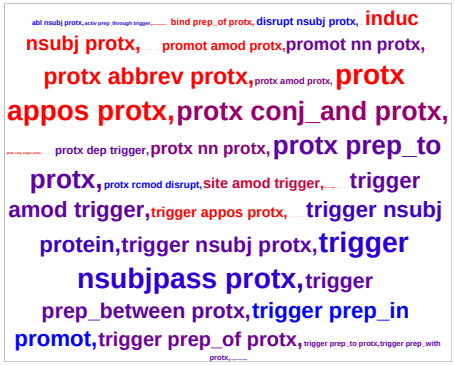


Figure 4.13: The most discriminative lexical vertex walks for double-argument binding events.

dictionary look-up to identify synonyms and lexical variants could prove to be of value.

Further analysing other lexical patterns of the phosphorylation trigrams, the pattern 'phosphoryl of $protx$' indicates a strong positive, while 'phosphoryl by $protx$' leads to negative events. While this seems counter-intuitive at first sight, it can be explained by taking the definition of the phosphorylation event into account: the argument of the event should always be the protein that is phosphorylated. In terms of the BioNLP'09 Shared Task data, the pattern 'phosphoryl by $protx$' would lead to a regulatory event in which a protein regulates phosphorylation of yet another protein (Section 4.1.2). The classifier thus labels these correctly as negatives in the phosphorylation dataset.

Finally, interesting linguistic patterns can be found when analysing the tag cloud of lexical vertex walks in the dataset on double-argument binding (Figure 4.13). When a direct link exists between the two GGPs involved, this strongly points to a negative example (e.g. '$protx$ conj_and $protx$' or '$protx$ abbrev $protx$'). On the other hand, the nature of the link between a trigger and a GGP is highly informative (e.g. '$trigger$ prep_between $protx$' or '$trigger$ nsubj $protx$'). Most of the highly ranked vertex walks involve nodes that have been blinded, confirming the usefulness of the blinding step to improve generalization (Section 4.2.4).

4.5 Conclusion

We have participated in the BioNLP'09 Shared Task on Event Extraction, following a community-wide shift from relation-based extraction towards the extraction of events from biomolecular texts. Out of the 24 international participants, we achieved the 5^{th} place with an overall F-score of 40.54%. In particular, our system accurately extracts gene expression, protein catabolism and phosphorylation events, while binding and regulatory events are more challenging.

In this chapter, we have discussed and benchmarked several important design choices for an event extraction framework, ranging from various text pre-processing methods and parameter optimization to the modularity of the system and size of the training data. We noticed that the task of extracting biomolecular events leads to high-dimensional and unbalanced datasets and thus carefully designed our system in order to improve balance of the datasets and to avoid false positives. Both learning curves as well as precision-recall curves have shown interesting characteristics related to this specific challenge. These extensive analyses have led to a relative performance gain of 10%. Our system now achieves 37.43% recall, 54.81% precision and 44.48% F-score.

Finally, we have introduced the application of ensemble FS to event extraction. Thorough analyses have shown that this robust FS method is well suited to tackle text mining challenges, eliminating up to 90% of all features before performance drops below the baseline without feature selection. Classification improves most when eliminating 75% of all features, considerably reducing dimensionality of the datasets. We have additionally shown that our ensemble FS approach provides insight into the predictions of the black-box model of machine learning methods. Analysis of the top selected features has illustrated various interesting patterns, both in terms of biology and from a linguistic point of view.

5

Entity relations

During the past decade, biomedical text mining tools have evolved from extracting simple binary relations between genes or proteins (Chapter 3) to a more expressive event representation (Chapter 4). These binary relations or events always pertain to GGPs, i.e. genes, proteins and mRNA. A recent challenge now targets relations between GGPs and a broader category of entities, covering domain terms that can not be annotated as named entities (NEs), but that are still highly relevant for biomedical information extraction (Ohta *et al.*, 2009). In contrast to relations involving change or causality, these relations model static hierarchies involving Equivalence (equal-to), Member-Collection (subunit-of) or Protein-Component (part-of). These non-causal relations are termed '*entity relations*' (REL), or, in previous work, '*static relations'* (Pyysalo *et al.*, 2009).

A more detailed explanation of the REL data and its applications is given in Section 5.1. We then detail a study on integrating REL data with events to refine and improve event extraction, using gold-standard REL annotations (Section 5.2). Next, we describe an extension of our machine learning framework for the prediction of entity relations, applying semantic lexicons and automatically derived semantic similarities between domain terms (Section 5.3). Further, we analyse the performance and strengths of both our own framework as well as the best performing system of the BioNLP Shared Task 2011, and create a hybrid system combining the two. Finally, we use the ensemble feature selection method presented in Chapter 4 to analyse and visualise the most discriminative patterns in the dataset on entity relations.

Type of relation	Examples
Equivalence	[human *interleukin 2* gene] [*TNF-alpha* mRNA transcripts]
Member-Collection	[*Ikaros* family members] [inflammatory cytokine genes] including *TNF*, *IL-1*, and *IL-6*
Protein-Component	[*alpha globin* regulatory element] [tyrosine] phosphorylation of *STAT1*
Subunit-Complex	[*Myc-Max* heterodimer] *p50* or *relA*, the two major subunits of [NF-kappaB]

Table 5.1: Examples of entity relation types, including both embedded and non-embedded cases. GGPs are in italic and domain terms are delimited by square brackets.

5.1 Introduction

5.1.1 Applications

The research domain of systems biology has emerged from the insights that the behaviour of a system can not be explained by only analysing its parts ('reductionism'); instead it is necessary to study the system as a whole ('holism'). Consequently, data integration and modelling of complex networks are at the heart of systems biology approaches and have become more and more common practice. Following this trend, the BioNLP field is also moving towards modelling more complex interactions than ever before. The shift of focus from binary relations to an event representation (Chapter 4) is one example, and the study on entity relations, presented in this chapter, is another.

Entity relations are non-causal relations between a GGP symbol (e.g. *Esr-1*) and a domain term. Domain terms are usually general words denoting biomolecular concepts such as 'promoter' or 'complex'; occasionally such concepts have a specific name such as *NF-kappaB*. A few examples of entity relations are depicted in Table 5.1. Typically, they express a relationship between two molecular entities without necessary implication of causality or change.

Entity relations provide the opportunity to further refine text mining results beyond the event representation. To illustrate, Figure 5.1 depicts a sentence containing both events as well as entity relations. The event annotation indicates an expression event involving the GGP *interleukin-3*. Regulation of this gene expression event is stated by the trigger word 'mediated'. In addition, the REL annotation marks two terms that refer to parts of the GGP, namely 'cis-acting elements' and 'transcription starts'. These two domain terms provide more detailed information and by combining the two types of annotation, a more extensive representation of the extracted information is provided. This

Figure 5.1: A sentence from PMID:8662845, showing how the event dataset (green, single line) and the REL dataset (orange, double line) offer complementary information.

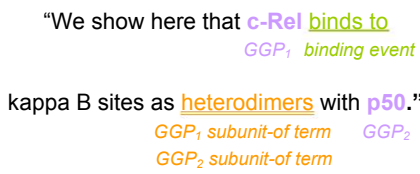


Figure 5.2: A sentence from PMID:1372388, showing how REL data (orange, double line) can provide strong clues for the extraction of biomolecular events (green, single line) from text.

could be particularly useful for applications such as abstract summarization or integration of the predictions into complex regulatory pathways.

In addition to providing an enhanced representation of biological processes, entity relations also offer interesting opportunities to improve the event extraction algorithms. As an example, consider the sentence presented in Figure 5.2, in which *c-Rel* and *p50* are both annotated as subunits of the term 'heterodimers'. The REL data thus provides strong clues for the extraction of a binding event between *c-Rel* and *p50*.

Further, entity relations may be helpful in pruning false-positive event predictions. Consider, for example, the statements 'GGP_1 *binds* GGP_2 *promoter*' and 'GGP_1 *binds* GGP_2 *inhibitor*': a binding event involving GGP_1 and GGP_2 should be extracted for the first statement, but not for the second.

Finally, an important application of entity relations is to be found in gene name normalization systems, which aim at linking ambiguous, text-bound gene symbols to unique, external database identifiers (Section 1.3.4). For example, *BIRC3* maps to Entrez Gene ID 330[1], and the full term *human BIRC3 gene* can be linked to the same unique identifier. However, the phrase *the BIRC3 inhibitor* refers to an entirely different molecular entity. By formally defining these relations, a text mining module is able to establish semantic links between various molecular entities found in text (e.g. inhibitors, promoter constructs, gene families, etc.).

[1] `http://www.ncbi.nlm.nih.gov/gene/330`

Relation type	Training	Testing
Protein-Component (ST)	1689	334
Subunit-Complex (ST)	751	163
Equivalence (GENIA - EB)	720	129
Functional (GENIA - EB)	110	17
Locus (GENIA - EB)	11	5
Member-Collection (GENIA - EB)	5	0
Misc (GENIA - EB)	53	11
Object-Variant (GENIA - EB)	14	5
Out-of (GENIA - EB)	40	7
Protein-Component (GENIA - EB)	222	51
Subunit-Complex (GENIA - EB)	108	22
Member-Collection (GENIA - NEB)	760	181
Protein-Component (GENIA - NEB)	593	174
Subunit-Complex (GENIA - NEB)	275	82

Table 5.2: Number of positive instances of the various types in the entity relation corpora. ST refers to the Shared Task data, while GENIA refers to the GENIA relation corpus. The latter corpus is further divided into embedded (EB) and non-embedded (NEB) cases.

5.1.2 Corpora

There are two related corpora publicly available with annotations for entity relations: the data of the BioNLP ST of 2011 and the more extensive GENIA relation corpus. The characteristics of these two corpora are summarized in Table 5.2 and Table 5.3. The ST'11 data fully corresponds to the dataset of the ST'09 (Section 4.1.4), and is divided into three distinct datasets: training (800 abstracts), development (150 abstracts) and test data (260 abstracts) (Kim *et al.*, 2011). The training set of the GENIA relation corpus corresponds to the training set of the ST data, and the test data corresponds to the ST development data. In both corpora, valid entity relations involve exactly one GGP and one domain term and both occur within a single sentence. Gold-standard relations are provided for the training and development set, allowing the application of machine learning algorithms to produce predictions for the test set.

The ST'11 data defines two types of entity relations. A Subunit-Complex relation holds between a protein complex and its subunits, while a Protein-Component relation is less specific and involves a GGP and its components, such as protein domains or gene promoters. The GENIA relation corpus contains several other types, including Equivalence and Member-Collection, which expresses a relationship between e.g. a gene family and its members. This corpus is further split into 'embedded' and 'non-embedded' relations, the first being relations occurring within a noun phrase (Ohta *et al.*, 2009), and the latter containing broader relations between nominals (Pyysalo *et al.*, 2009).

	Relation types	Embedded distinction	Gold GGPs	Gold terms	800 articles	150 articles	260 articles
ST	2	no	yes	no	train	dev.	test
GENIA	9	yes	yes	yes	train	test	-

Table 5.3: Characteristics of the two different REL corpora.

5.2 Integration with event predictions

This section describes a thorough study on how entity relations can be integrated into the event extraction framework described in Chapter 4. In this theoretical analysis, gold-standard annotations from the GENIA relation corpus are used and integrated with the event framework, benchmarked on the ST'09 event corpus (Section 4.1.4). The development set (150 articles) is used as test set for evaluating these analyses as the gold-standard REL/event annotations on the final test set (260 articles) are hidden.

To simplify the analysis, we further focus on physical, non-regulatory events involving only the given GGPs as participants. The inclusion of regulatory events would introduce a number of complications for evaluation, as failure to extract a referenced event implies failure to extract events in which they appear as arguments (Section 4.3.1). Even with these limitations, the data still contains over 800 development test events for use in the analysis. For the REL data, all relation types as listed in Table 5.2 are used.

First, the number of useful complementary annotations across both datasets is analysed (Section 5.2.1). Next, we describe the generation and evaluation of new candidate events using terms involved in entity relations, in an effort to boost recall of the event predictions (Section 5.2.2). To additionally improve on precision, we have implemented a false positive filter exploiting REL annotations of GGPs involved in relations judged to serve as negative indicators, such as 'GGP inhibitor' (Section 5.2.3). Finally, Section 5.2.4 details experiments on the creation of a more extensive feature set for event extraction by including entity relation data.

5.2.1 Complementary data

To assess the usability of the REL data for event extraction, we first analyse the number of complementary annotations across the two datasets. On the document level, there is at least one REL annotation for 87.6% of all training set articles and for 94.67% of the development test set, including both positive entity relations as well as explicitly negated ones. Most articles from the event dataset thus involve GGPs at least potentially involved in entity relations.

Analysing the overlap in more detail, we determined the number of events that could benefit from adding REL data by counting the number of events for which at least one GGP is also involved in an entity relation (positive or negative). Table 5.4 shows the

Events	Training		Dev. test	
Pos. REL data	1190	32%	227	28%
Neg. REL data	841	22%	207	26%
All REL data	1635	44%	350	43%

Table 5.4: Number of events that can be linked to at least one entity relation, including explicitly annotated negative annotations.

Event type	Instances	Max. rec.
Gene expression	63	17.70%
Transcription	34	41.46%
Protein catabolism	4	19.05%
Phosphorylation	20	42.55%
Localization	4	7.55%
Binding	73	29.44%
All events	198	24.54%

Table 5.5: Maximal recall performance of event instances involving at least one domain term as argument. These terms are functioning as aliases for the GGPs they are positively associated with.

results of this assessment. In the training data, 1635 events involve at least one GGP with REL annotation, which is 44% of all events in the gold-standard annotation. For the development test set, the number is 350 out of the 808 gold-standard events (43%).

5.2.2 Domain terms as aliases for related GGPs

The first application of entity relations in an event extraction framework involves the use of domain terms appearing in the REL data as aliases for the GGPs they are positively associated with. In the event extraction framework, new candidate events can thus be formed by treating the terms as GGPs, and mapping them back to the real GGPs after classification. This procedure is motivated by the definition of the various REL types and the underlying biological processes. For example, if a complex is known to activate the expression of a certain target GGP, then the various subunits of this complex can be annotated as participants in that event.

Obviously, this approach has some intrinsic limitations as not all GGPs occurring as arguments in events have a corresponding term that could be used as alias. To assess the maximal recall, the original event extraction framework is employed (Section 4.2), removing the SVM classifier from the pipeline and simply labeling all newly constructed candidate events as positive predictions. The result indicates that the framework is capable of retrieving 198 gold-standard cases (Table 5.5), which is 24.54% of all events. Some missing events may involve trigger words not included in the dictionary (Section 4.2.2), preventing the event to be formed as a candidate, but most events can not

Event type	r	p	F
Gene expression	11.24	81.63	19.75
Transcription	20.73	89.47	33.66
Protein catabolism	19.05	100.00	32.00
Phosphorylation	36.17	100.00	53.12
Localization	3.77	25.00	6.56
Binding	12.50	45.59	19.62
All events	13.75	67.27	22.84

Table 5.6: Performance of event instances involving at least one domain term as argument. These terms are functioning as aliases for the GGPs they are positively associated with.

be reconstructed because not all their GGP-arguments are positively linked to a domain term through a REL annotation.

While the results show that nearly 25% of all events are potentially retrievable by using domain terms as aliases for GGPs, this percentage varies greatly across event types. For example, less than 8% of localization events can be found with this scheme, while maximal recall for phosphorylation events is over 40%. These results reflect the intrinsic differences between event types and the ways in which they are typically expressed, and suggest that it should be beneficial for event extraction to take these differences into account when incorporating entity relations.

Having established an upper bound for recall, a subsequent experiment involves treating the newly created instances as normal candidate events. For classification, an SVM is trained on regular candidate events involving GGPs (Section 4.2.5), as this ensures sufficient training material.

Both lexical and syntactic patterns are expected to be similar for events involving either domain terms or GGPs. To test this hypothesis, the event-extraction pipeline is employed for these new instances. Evaluation is performed with the standard evaluation script provided by the BioNLP'09 Shared Task organizers and the results are detailed in Table 5.6. While we have already established that recall is subject to severe limitations (Table 5.5), we note in particular the high precision rates of the predictions. In particular, four out of six event types achieve a precision rate higher than 80%.

To allow for a meaningful comparison, these results should be put into perspective by merging the new predictions with the predictions of a baseline extractor and comparing against this baseline[2] (Table 5.7). This analysis reveals interesting results: while overall performance increases slightly from 64.71% to 65.33% F-score, this trend is not common to all event types. For instance, prediction of localization drops 3.23 percentage points in F-score. Considering the maximum recall results, this is not entirely surprising and confirms that the prediction of localization events does not benefit from entity relation data in this approach.

[2]These numbers vary slightly with those reported in Chapter 4 because the classifiers were rebuilt.

Event type	Baseline	Merged
Gene expression	77.01	77.56
Transcription	63.41	64.24
Protein catabolism	86.36	86.36
Phosphorylation	70.10	76.47
Localization	80.00	76.77
Binding	38.69	40.52
All events	64.71	65.33
All events (precision)	69.11	67.19
All events (recall)	60.84	63.57

Table 5.7: Performance of the event extraction framework. First column: using only normal events involving GGPs ('baseline'). Second column: merging the new predictions (Table 5.6) with the first ones. All performance rates indicate F-score, except for the last two rows.

However, we do observe a considerable increase in performance for phosphorylation events (6.37 pp. in F-score) and some increase for binding events (1.83 pp. in F-score). These effects are mainly caused by an increase in recall (10.64 and 4.43 pp., respectively). When considering all event types, recall increases from 60.84% to 63.57% (Table 5.7, last row). These results clearly indicate that the inclusion of entity relations can improve recall while retaining and even slightly improving general performance.

5.2.3 Filtering false positive events

To further improve event extraction performance, a false-positive filter was implemented using specific categories of relations serving as negative indicators for event extraction. In particular, the 'Out-of' category and explicit negative instances of the 'Functional' relation annotations were used, covering instances like 'GGP inhibitor'. In such cases, the FP filter prohibits the embedded GGP to participate in any event as the topic of the sentence is not the GGP itself, but e.g. its inhibitor.

In the development test set, this strategy has automatically identified 24 relevant GGP mentions that should not be annotated as being involved in any event. Even though this number is relatively small, we aim at designing a high specificity FP filter while relying on the SVM classifier to solve more ambiguous cases.

Applying the FP filter to the baseline result detailed in Table 5.7, three events are discarded from the set of predictions. All three instances represented false positives: two binding events and one gene expression event. Overall precision and F-score increased marginally by 0.30 and 0.13 pp., respectively.

5.2.4 Extended feature representation

The last type of experiment aims to boost both precision and recall by substantially extending the feature generation module for event extraction using the newly introduced REL data. Table 5.4 shows that such an enhanced feature representation could influence 1635 events in the training data and 350 events in the development test data, covering a significant part of the dataset (43-44%).

Building further on the feature generation module for event extraction as described in Section 4.2.4, a range of new features is added to the feature vectors while also providing enhanced generalization of existing features. Generalization is crucial for the text mining framework as it enables the extraction of relations from new contexts and forms of statements.

First, for each domain term involved in an entity relation, the string of the term is included as a separate feature. This generates relation-associated features such as 'tyrosine', which is strongly correlated with phosphorylation events. For terms spanning multiple tokens, each token is additionally included as a separate feature, capturing commonly used words such as 'promoter' and 'receptor'. Each distinct feature is linked to its specific relation type and indicates whether it was derived from a positive or negative REL annotation.

Additionally, a new feature type was introduced that expresses whether or not the trigger of the event is equal to a domain term related to one or more GGPs involved in the event. To illustrate the relevance of such a feature, consider the trigger 'homodimer'. If the GGP involved is annotated as being a subunit of this homodimer, this provides a strong clue for the extraction of a binding event. Similarly, the explicit negation of the existence of any entity relation indicates a negative event.

Apart from adding these new features, we have used the entity relations to generalize the lexical patterns. In particular, the lexical information in the feature vector was generalized by blinding terms involved in entity relations. For each such domain term, the whole string is replaced by one word, expressing the type of the entity relation and whether the relation is positive or negative. This results in more general patterns such as 'inhibit prep-to $partx$' (vertex walk) or 'activ in $nonpartx$' (trigram). In Figure 5.2, 'heterodimer' would be blinded as $complexx$ as both *c-Rel* and *p50* are members of this complex.

Initial experiments with the extended feature representation show that a small increase in performance could be obtained on the development test set, achieving 61.34% recall, 69.58% precision and 65.20% F-score. However, it also became clear that not all event types benefit from the new features. Surprisingly, binding is one such example. We hypothesize that this is mainly due to the intrinsic complexity of binding events, requiring an even more advanced feature representation.

To take the inherent differences between various event types into account, the optimal set of features was selected for each type. In a new experiment, the feature generation step thus depends on the event type under consideration. Table 5.8 details the

Event type	Baseline	Extended
Gene expression	77.01	78.06
Transcription	63.41	63.80
Protein catabolism	86.36	86.36
Phosphorylation	70.10	76.29
Localization	80.00	84.21
Binding	38.69	38.34
All events	64.71	65.73
All events (precision)	69.11	69.99
All events (recall)	60.84	61.96

Table 5.8: Performance of the event extraction framework. First column: using the baseline feature representation. Second column: using the extended feature representation. All performance rates indicate F-score, except for the last two rows.

results of this optimization: an overall F-score of 65.73% is achieved. Similar to the experiments in Section 5.2.2, the F-score for the prediction of phosphorylation events increases by 6.19 pp. Additionally, an increase of 4.21 pp. in F-score is obtained for localization events, even though we were unable to improve on them when using terms as aliases for additional candidate events (Section 5.2.2). Additional experiments suggest that this is because the prediction of localization events in general does not benefit from positive entity relations, but negative entity relations do provide strong clues to the SVM classifier.

5.3 REL extraction framework

The previous section has shown promise for improving event prediction with entity relations, at least for a few specific event types. Additionally, the extraction of entity relations is a valid challenge in its own right, as these relations provide an enhanced level of detail for text mining systems. In this section, we investigate the performance of predicting entity relations, applying a new extension of our previously introduced machine learning framework.

The prediction of entity relations starts with a novel module designed to calculate semantic similarities between domain terms (Section 5.3.1). These similarities are used to construct generalized feature vectors that represent the semantic and grammatical information contained in the training sentences with GGPs. The rich feature vectors are then subjected to feature selection and subsequently used for training a binary SVM for each entity relation type (Section 5.3.2). Finally, for each selected sentence and each GGP occurrence, a suitable domain term is selected within a certain search window (Section 5.3.3). The flowchart of our framework is depicted in Figure 5.3.

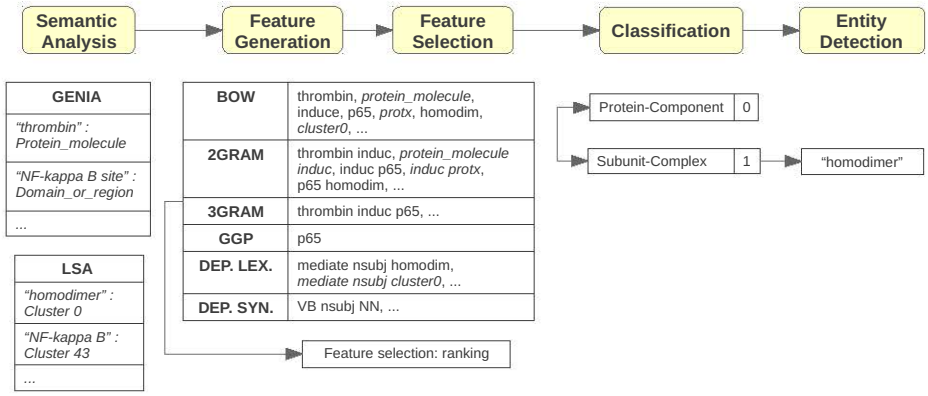


Figure 5.3: Flowchart of our framework, including example intermediate steps for the sentence 'Thrombin-induced *p65* homodimer binding to downstream NF-kappa B site of the promoter.'

5.3.1 Semantic analysis

To fully understand the relationship between a GGP and a domain term, it is necessary to account for the usage of synonyms and lexical variants in human language. Two strategies to capture this textual variation were implemented, creating useful semantic lexicons that group similar words together. The first method takes advantage of manual annotations of semantic categories in 1000 articles. The second method relies on statistical properties of nearly 15,000 articles.

GENIA term corpus

The GENIA term corpus contains manual annotations of various domain terms such as promoters, complexes and other biological entities (Kim *et al.*, 2008a). These annotations are used to link certain lexical patterns to semantic categories, such as *DNA-domain-or-region* and *protein-family-or-group*. The GENIA term corpus consists of the same 1000 abstracts as the combined training and development ST data, and is therefore a highly suitable additional data source.

Semantic spaces

In addition to using the GENIA term corpus, we have also calculated semantic spaces to deduce the underlying similarities between domain terms. A semantic space can be defined as a mathematical representation of a text corpus, containing high-dimensional vectors that capture the context in which certain words are used. Similar vectors then

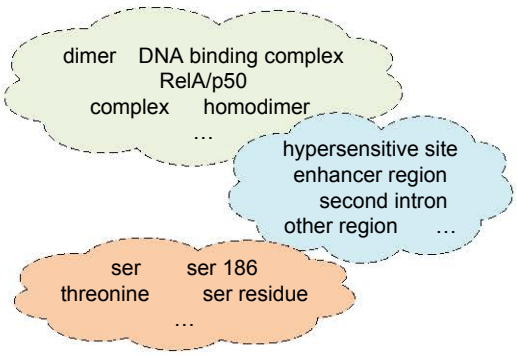


Figure 5.4: A few examples of clustered domain terms, obtained with LSA and MCL.

represent semantically similar words. By applying semantic spaces, we aim at finding clusters of closely related biomolecular concepts, such as *complex* and *heterodimer*.

In a first step, a large-scale corpus is collected, containing 14,958 PubMed articles on *human transcription factor blood cells*, which is the topic of the GENIA and ST corpora. This initial collection ensures coverage of most domain terms occurring in our dataset, while at the same time, the size of the collection guarantees that the results are sufficiently general to be applicable in other domains. All the words in the collection are transformed to their lowercase variants and the Porter stemming algorithm is used for generalization purposes (Porter, 1980).

The actual semantic spaces are then built with the open-source S-Space Package (Jurgens and Stevens, 2010). This package contains implementations of several different semantic algorithms that have been extensively documented, tested and validated. We have experimented with latent semantic analysis (LSA) (Landauer and Dumais, 1997), RI (Sahlgren *et al.*, 2008), HAL (Burgess and Lund, 1997) and COALS (Rohde *et al.*, 2006). By running these semantic algorithms on the nearly 15 thousand articles, we obtain datasets of relevant terms and their semantic vectors.

In a final step, these semantic vectors are clustered into meaningful groups. Clustering was performed using the Markov Cluster (MCL) algorithm (van Dongen, 2000) with the cosine similarity measure. MCL is a graph-based clustering algorithm that finds densely connected sub-graphs by using flow simulation. In this context, the nodes of the graph are the semantic vectors and the similarities determine the weights of the edges between these nodes. The inflation factor of MCL was optimized to obtain a good trade-off between the generalizability of the cluster and its purity. To this end, a linear search between inflation values 2 to 40 was performed and ultimately an inflation factor of 6 was selected for further analysis, using the score heuristic described below.

To assess the best fitting semantic algorithm for this specific classification task, a score heuristic S was implemented to evaluate the resulting clusters. This score depends

on the homogeneity, reliability and predictive value of the clusters.

Some terms in the clusters can be assigned a gold-standard classification label by looking at the training portion of the GENIA relation corpus. For example, the domain term 'complex' is always associated with a Subunit-Complex relation. The number of such gold-standard labels in each cluster is represented by $Known$ and a cluster's homogeneity (HG) expresses the internal agreement between these labels. The homogeneity is multiplied by the number of unlabeled test terms ($Unknown$) to assess the predictive value of the cluster. The reliability ($Reliability$) of the cluster further expresses the percentage of known labels versus predicted ones. Clusters with relatively more known labels are deemed to be more reliable, unless the labels are highly contradicting, which would result in less homogeneity and thus a lower score. The final score metric S calculates one score for each cluster, and a clustering result is scored as the sum of all clusters.

$$S = Unknown \times HG \times Reliability \tag{5.1}$$

$$Reliability = \frac{Known}{Known + Unknown} \tag{5.2}$$

Evaluation using the score heuristic S clearly indicated that the semantic algorithms RI and HAL produce less useful results than LSA and COALS. After manual inspection of the clusters, LSA was chosen as the preferred method to produce semantic vectors. Figure 5.4 depicts some of the resulting clusters.

5.3.2 Machine learning module

The ML component of the framework identifies entity relations by analysing lexical and grammatical patterns in sentences containing GGPs. The feature generation module as well as the classifier are built upon the modules described in Section 3.2 (PPI extraction) and Section 4.2 (event extraction), adding also 2-grams to the feature set. Further, domain terms from the semantic lexicons (Section 5.3.1) are blinded with their corresponding clusters or categories for additional, generalized features. For each generalization, a blinded and a non-blinded variant is included in the feature vector. A few example features are depicted in Figure 5.3, with generalized features in italic. The final feature vectors are classified using an SVM with a radial basis function as kernel.

As for event extraction (Section 4.4), ensemble feature selection was performed to gain a better insight into the task at hand. For example, Figure 5.5 shows the feature cloud of the most informative feature patterns when predicting embedded Protein-Component events. Features indicating positive examples (blue) include words of the semantic class 'protein-domain-or-region' and the trigram 'human protx promoter'. Negative features (bright red) include the 2-gram 'protx subunit' and the semantic class 'protein-complex', which would in turn be a positive pattern for the Subunit-Complex

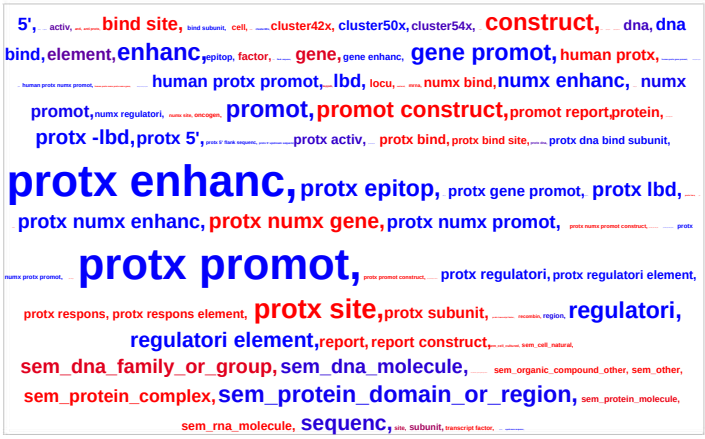


Figure 5.5: Most important features for predicting (embedded) Protein-Component relations, as predicted by our framework. The feature cloud shows all types of grammatical and lexical features that are most discriminative according to the ensemble feature selection algorithm. Red indicates features that mark negative examples, blue features mark positive examples.

type. Notably, there are almost no syntactic features in the top most informative features. This is a property inherent to the prediction of embedded entity relations, for which the close lexical context of the GGP is the most determining factor. In contrast, the non-embedded types do rely more on the syntactic structure of the sentence.

5.3.3 Term detection

In our framework, sentences are selected for classification if they contain at least one GGP. When the sentence is classified as containing a certain type of entity relation, it is necessary to also identify the exact domain term that is related to the GGP. To this end, a pattern matching algorithm was designed that searches within a given window (number of tokens) around the gene symbol. Starting by analysing the closest tokens, the window size is increased as long as a given number of domain terms has not been found. More specifically, at most one term is searched for the Subunit-Complex relation within a window of maximum 9 tokens, while for the Protein-Component, up to 3 terms are searched within a window of maximum 15 tokens. These parameter settings were empirically determined on the training data.

Within the search window, a rule-based algorithm decides whether a given token qualifies as a relevant domain term. To this end, a dictionary of high-precision domain terms was automatically assembled from the training data (e.g. ‘nf-kappa b complex’ or ‘binding site’). When there is no match, the algorithm tries to link the token to a semantic cluster as obtained by the LSA algorithm (Section 5.3.1). For this step, only clusters that

could be unambiguously linked to one specific type of entity relation (measured on the training data) were used. When this step fails too, finally a high-recall dictionary is used. This dictionary contains domain terms tagged in positive examples of the training data, removing non-frequent or too general terms.

This algorithm produces a maximal recall of 91% using an all-true classification on the development set, and was thus judged to be sufficiently able to identify domain terms in sentences that were likely to express an entity relation.

5.4 Results

In this section we first present the official performance results of our entity relation prediction framework on the Shared Task of 2011 (Section 5.4.1). We then explain the 16 pp. performance discrepancy between our own framework, ranking second, and the winning system of the ST'11, developed in Turku, by benchmarking on the GENIA relation corpus (Section 5.4.2). Subsequently, a hybrid framework is evaluated on the (hidden) ST test set. Finally, we experiment with further combinations of the frameworks and achieve either high-precision or high-recall results (Section 5.4.3).

5.4.1 Official results of the ST'11

The REL supporting task (Pyysalo *et al.*, 2011) of the BioNLP Shared Task of 2011 (Kim *et al.*, 2011) was specifically focused on extracting entity relations, contributing to the general goal of the ST to support more fine-grained text mining tools. The REL challenge involved the prediction of two types of entity relations: Subunit-Complex and Protein-Component (Section 5.1.2). The combined training and development sets included 751 positive examples of Subunit-Complex relations, and 1689 for the Protein-Component type.

For the evaluation on the final test set of the ST'11 data, the domain terms necessary to infer meaningful entity relations have to be detected by the participants as part of the challenge. Valid entity relations involve exactly one GGP and one domain term and such relations always occur within a single sentence. Gold-standard relations are provided for the training and development set.

Table 5.9 depicts the performance of the official submissions for the REL subtask of the Shared Task 2011. The system developed by the university of Turku obtained a first place with an F-score of 57.71% (Björne and Salakoski, 2011). Our system achieved a global performance of 41.62% F-score (Van Landeghem *et al.*, 2011a), ranking second. Concordia University ranked third with 32.04% F-score (Kilicoglu and Bergler, 2011). Finally, the University of Science (UoS), VNU, achieves 18.74% F-score.

The winning system, the Turku Event Extraction System (TEES), is a generalized biomedical relation extraction tool based on a unified, extensible graph representation,

	Subunit-Complex			Protein-Component			All		
	p	**r**	**F**	**p**	**r**	**F**	**p**	**r**	**F**
Turku	66.95	48.47	56.23	68.57	50.90	58.43	68.04	50.10	**57.71**
Ghent	38.12	47.85	42.43	36.53	47.31	41.23	37.04	47.48	**41.62**
Conc.	39.81	26.38	31.73	52.05	23.35	32.24	46.85	24.35	**32.04**
UoS	66.67	4.91	9.14	21.63	20.96	21.29	23.26	15.69	**18.74**

Table 5.9: The official performance on the ST test set, measured by precision, recall and their harmonic mean, the F-score (F).

where word entities (e.g. GGPs and event triggers) constitute the nodes and event arguments the edges (Björne and Salakoski, 2011). The system consists of a pipeline of three main components based on SVMs. With both frameworks based on machine learning, TEES uses a global approach, extracting all event types in a sentence at once, while our framework uses a parallellized approach (Section 4.2).

The term detection component of TEES is also SVM-based, using mainly features extracted from dependency parsing, word tokens and part-of-speech tags. The term detection algorithm of our own system however, is rule-based (Section 5.3.3).

The relatively high performance of TEES is remarkable, as this system has not been developed specifically for the detection of entity relations, but rather is able to generalize quite well to different text mining challenges (Björne and Salakoski, 2011). In contrast, our own framework contains specific algorithms designed for the REL classification task such as the creation of the semantic lexicons (Section 5.3.1). In the next section, we aim at elucidating the performance discrepancy between these two systems, by analysing whether most errors originate from the term recognition step or from the relation extraction module.

5.4.2 Analysis on the GENIA relation corpus

To analyse the 16 pp. performance discrepancy between the best ST result (TEES) and the second one (our system), a number of analyses were performed on the GENIA relation corpus (Van Landeghem *et al.*, 2012b). This corpus was chosen for two main reasons. First, its scope is broader in comparison to the ST data, as the annotations in the GENIA relation corpus cover several additional types of entity relations (Section 5.1.2). Second, the availability of gold-standard domain annotations in the GENIA relation corpus allows for benchmarking only the relation extraction module. This also means that the results obtained here are not directly comparable to the results on the ST data, because the latter corpus does not include gold-standard domain terms.

For these new analyses, TEES remained unchanged, while the feature generation of our own framework was modified slightly to benefit from the additional features of the GENIA relation corpus, optimizing feature sets for the different entity relation types

	TEES			Ghent		
Relation type	**Prec.**	**Rec.**	**F**	**Prec.**	**Rec.**	**F**
Equivalence (EB)	93.13	95.31	94.21	97.64	96.12	96.88
Protein-Component (EB)	96.08	96.08	96.08	100.00	86.27	92.63
Subunit-Complex (EB)	79.17	86.36	82.61	80.00	72.73	76.19
All (EB)	92.23	94.53	**93.37**	97.85	90.10	**93.81**
Member-Collection (NEB)	81.44	75.14	78.16	71.73	75.69	73.66
Protein-Component (NEB)	87.77	67.03	76.01	73.33	83.63	78.14
Subunit-Complex (NEB)	81.54	64.63	72.11	73.24	63.41	67.97
All (NEB)	83.83	69.89	**76.23**	72.65	76.50	**74.52**
ALL (EB+NEB)	86.83	77.55	**81.93**	79.94	80.82	**80.38**

Table 5.10: Performance on the GENIA relation corpus for embedded (EB) and non-embedded (NEB) relation types. Only showing results for datasets that are sufficiently large for application of ML techniques (Section 5.1.2).

(embedded vs. non-embedded). Due to the available gold-standard terms, our framework now classifies sentences with exactly one GGP and one term, rather than just sentences containing one GGP. Consequently, additional features describing the lexical and semantic content of the tagged domain terms are added to the feature vectors.

The parameters for the TEES classifier have been optimized on the ST development corpus, which roughly corresponds to the test set of the GENIA relation corpus. For our own framework, the best feature set was selected after several analyses on the same dataset. These settings result in slightly optimistic performance values for both systems, when benchmarking on the GENIA relation corpus. However, the resulting overfitting only accounts for a few percentage points in F-score, and because these analyses are used for comparison between TEES and our own framework, this is not considered to be a problem. This is even more the case because the hidden ST test set is the *de facto* standard for benchmarking and comparing different systems. The results on this dataset are described in the next section.

For the classification experiments on the GENIA relation corpus, separate runs were performed for 'embedded' and 'non-embedded' relations. The performance results are depicted in Table 5.10. From this table, we learn that both frameworks perform almost equally well, with a small advantage of TEES. The huge discrepancy, as observed in the official results, has disappeared. This can be explained by the availability of the gold-standard domain terms, but may also be due to the added relation types. For example, our framework performs worse than TEES for the Subunit-Complex relation type in both the ST and GENIA evaluations, but performs better for the Equivalence type, which is not included in the ST evaluation. In the next section, we will further analyse the influence of the term detection module by creating a hybrid framework.

Another important result emerging from the analysis on the GENIA relation corpus, is the performance discrepancy between embedded and non-embedded types. For the

	Subunit-Complex			Protein-Component			All		
	p	**r**	**F**	**p**	**r**	**F**	**p**	**r**	**F**
Turku	66.95	48.47	56.23	68.57	50.90	58.43	68.04	50.10	**57.71**
Ghent	38.12	47.85	42.43	36.53	47.31	41.23	37.04	47.48	**41.62**
Hybrid	66.95	48.47	56.23	61.79	52.40	56.70	63.32	51.11	**56.56**
T ∩ H	75.25	46.63	57.58	71.56	48.80	58.03	72.70	48.09	**57.89**
T ∪ H	60.74	50.31	55.03	59.73	53.89	56.66	60.05	52.72	**56.14**

Table 5.11: Performance on the ST test set, measured by precision, recall and their harmonic mean, the F-score (F). The first few rows indicate the official results. Next, the performance of the hybrid system is shown. Finally, the two last rows report on the performance of creating the intersection and the union of Turku's TEES system (T) and the hybrid (H) system.

embedded cases, global performance reaches around 93-94% F-score, while the non-embedded relations are predicted with an average F-score of 74-77%. The embedded cases are indeed less grammatically complex than the non-embedded ones. Interestingly, they do represent an important sub-challenge of entity relations. When combining text mining results with public databases, automatically tagged symbols need to be resolved to the correct record in the database. Such symbols are often extracted by named entity recognition software such as BANNER (Leaman and Gonzalez, 2008), which applies statistical models for the recognition of GGP symbols in text, and might sometimes tag a whole noun phrase rather than just the embedded GGP name. Embedded relation types formally describe the relationship between e.g. *Esr-1* and *Esr-1 promoter*, thus providing an automatic way of dealing with these strings and enabling a meaningful integration between text and database records.

5.4.3 Combining two frameworks

To test the hypothesis that our framework lags behind because of its term detection module, a hybrid framework was created by combining the term detection module of TEES with our relation detection module. This new, hybrid framework is tested on the official ST test data and it performs almost equally well as the original TEES submission (1.15 pp. lower F-score, Table 5.11). This result clearly shows the huge impact of the term detection module on the final results, as the relation extraction modules perform almost equally well. Apparently, the SVM-based term detection module of TEES performs much better than our rule-based approach, resulting in a much higher global performance result on the ST data.

Even though the performance of TEES and the hybrid framework are similar, there is still a considerable variability in the underlying predictions, as the relation extraction component differs significantly. Because of this, we can experiment with ensemble methods to combine both systems. Considering we only have access to two systems, the options for creating combinations are limited.

First, we have created the intersection of the two systems. Comparing two relations across the different frameworks is straightforward because they use the same GGP occurrences (gold-standard annotations) and the same domain terms (predicted by TEES). The results are shown in Table 5.11. Obviously, an intersection could never improve on recall compared to the original TEES submission, but we do find a precision increase of 2.99 and 8.30 pp. for Protein-Component and Subunit-Component respectively. The resulting F-score is 0.19 pp. higher, marginally better than the original TEES submission. However, the difference is not statistically significant, and this new framework is also more complex as it needs to train two different classifiers. Finally, it is important to note that any machine learning framework can in theory be tuned to achieve either high recall or high precision by applying the well-known precision-recall trade-off (Section 1.3.8).

The union of TEES and the hybrid system was subsequently constructed aiming at higher recall rates while still benefiting from the relatively high precision rates of both systems. However, this approach seems to include many irrelevant false positives (Table 5.11, last row). Recall rises with 2.99 and 1.84 percentage points for Protein-Component and Subunit-Component respectively, but F-score drops with 1.57 percentage points compared to the original TEES submission.

5.5 Conclusion

Data on entity relations not only offers a more detailed representation of biomolecular events, but can also help to boost the performance of event prediction. We have presented the first study on the applicability of entity relations for improving event prediction in biomedical texts. To investigate these opportunities, three sets of experiments were performed using gold-standard REL annotations. First, we have designed new candidate events by treating domain terms as aliases for the GGPs they are possitively associated with. By augmenting the normal event predictions with predictions for these new candidates, we have established a considerable increase in recall. Next, we have implemented a false positive filter to improve precision, by exploiting annotation for relations judged to imply only distant associations of the GGP and the enclosing noun phrase. Finally, the last type of experiment involves integrating complementary data on entity relations to obtain more informative feature vectors for candidate events. Results show that both recall and precision can be increased slightly by this last, more complex configuration.

During the experiments, it has become clear that there are important differences between the distinct event types. For example, phosphorylation events benefit most from adding REL data (increase of 6.37 pp. in F-score), while localization events can be enhanced using only features of negative REL annotations (increase of 4.21 pp. in F-score). For some event types, such as protein catabolism, the current techniques for integration of entity relations do not generate a performance boost at all. This is not a surprising result, as protein catabolism events, as defined in the BioNLP ST and often described in text, do not pertain to static domain terms as is the case for e.g. phosphorylation (site

argument) and localization (current/target location) (Section 4.1.1).

To predict entity relations rather than using gold-standard relations, an extraction system was implemented, based on previous work on PPI and event extraction. The REL framework consists of an additional clustering component that groups semantically similar words together using semantic spaces, as well as a rule-based component that detects the domain terms in these sentences. In the official BioNLP Shared Task 2011, our system ranked second with an F-score of 41.6%.

Finally, we have analysed the 16 pp. performance discrepancy of our framework with the best ranking system developed in Turku, TEES. Benchmarking on a related and more extensive dataset has guided the construction of a hybrid framework which combines the TEES term recognition module with our relation detection module. From these experiments, it became clear that the term detection module has a much higher impact than the relation extraction module on the final performance, and future development efforts in this field should thus focus more on accurate detection of the domain terms.

6

EVEX: Mining the bibliome

In the previous chapters, we have described several theoretical text mining studies on extracting protein-protein interactions (Chapter 3), biomolecular events (Chapter 4), and non-causal entity relations (Chapter 5). These algorithms have been benchmarked on manually annotated datasets consisting of hundreds of abstracts. However, to be useful in real-life scenarios, text mining tools need to be applied on a much larger scale.

In this chapter, a large-scale text mining resource is presented, providing detailed event-based representations of bibliome-wide biological statements. The bibliome entails all scientific literature on biological or biomolecular studies. Specifically, this study includes all PubMed abstracts and PubMed Central Open Access full-text articles (Section 6.1). The resulting dataset contains more than 67 million GGP occurrences and 34 million extracted events. To assess the out-of-domain performance of the text mining algorithms, a framework for manual evaluation was designed and employed for a subset of this data. The results are presented in Section 6.2.

One of the major limitations of the core text mining events, is that they are strictly text-bound and provide no facility for a more general treatment, such as being able to abstract from different name spelling variants and symbol synonymy. To resolve this issue, it is crucial to first produce canonical forms of the automatically tagged biological entities (Section 6.3.1). A gene symbol disambiguation algorithm then links these canonical forms to gene families (Section 6.3.2). A recent extension further normalizes gene symbols to unique Entrez Gene identifiers (Section 6.3.3).

On top of dealing with gene name ambiguity, several methods were designed to

	r	p	F
ST'09, abstracts	46.73	58.48	51.95
ST'11, abstracts	50.06	59.48	54.37
ST'11, full-text	48.31	53.38	50.72
ST'11, all	49.56	57.65	53.30

Table 6.1: Official performance of the event extraction component (TEES), evaluated on the ST'09 and ST'11.

refine and generalize the complex event patterns, for example by cleaning up long regulation chains (Section 6.4.1). Further, a pairwise view on top of the events was implemented (Section 6.4.2), as well as a module which finds indirect associations between GGPs (Section 6.4.3). Finally, the text-bound event occurrences are aggregated into generalized events by accounting for variation in both the gene symbols and the event structures (Section 6.4.4).

The resulting resource is distributed as a MySQL database[1] and forms a rich resource for homology-based hypotheses and literature-wide event retrieval (Section 6.5). For the text mining data to be truly useful in real-life cases, it needs to be accessible by researchers outside of BioNLP. To this end, we have built a publicly available web application[2] on top of the database (Section 6.6). Both the MySQL database and the web application are named EVEX, short for EVent EXtraction.

6.1 Core text mining predictions

The core text mining predictions in the EVEX dataset were produced in 2009 by the Turku Event Extraction System (TEES), the winning system of the BioNLP'09 Shared Task on Event Extraction (Kim *et al.*, 2009). TEES achieved 46.73% recall, 58.48% precision and 51.95% F-score (Section 4.3.2). This open-source extraction system[3] was combined with the BANNER named entity recognizer (Leaman and Gonzalez, 2008), trained on the GENETAG corpus of manually tagged PubMed abstracts (Tanabe *et al.*, 2005). In 2010, these two components together formed a complete event extraction pipeline with the highest reported accuracy.

Anno 2012, this pipeline remains state-of-the-art. Recent releases of BANNER are competitive with the best systems at the BioCreative 2 gene mention recognition task (Krallinger *et al.*, 2008). TEES further achieved the best performance for the REL sub-challenge (Section 5.4.1) and produced state-of-the-art results for various other challenges in the BioNLP'11 Shared Task. On the GENIA sub-challenge, which contains the

[1] `http://bionlp.utu.fi/pubmedevents.html`
[2] `http://www.evexdb.org`
[3] `http://bionlp.utu.fi/eventextractionsoftware.html`

event types as discusses in this work, TEES obtained 49.56% recall, 57.65% precision and 53.30% F-score (Björne and Salakoski, 2011; Björne *et al.*, 2012).

The file format and structured output of the pipeline correspond to the definition of the ST'09 (Section 4.1.1). In this chapter, events are stated using a simple bracketed notation, where the event type is declared first, followed by a comma-separated list of arguments enclosed in parentheses. Each argument consists of a GGP and is proceeded with C: and T:, denoting the role of the argument respectively as (C)ause or (T)heme. For instance, the negative regulation event depicted in Figure 4.1 (page 4-3) would be stated as

```
Negative-regulation(T:Binding(T:HIV-TF1)).
```

On top of the core events (task 1), the TEES system additionally extracts data relevant to tasks 2 and 3 of the ST'09. Task 2 is concerned with the extraction of additional entities such as cellular locations and phosphorylation sites, and task 3 deals with negative and speculative information in text (Section 4.2).

6.1.1 PubMed abstracts

Originally, the event extraction pipeline was applied to all citations in the 2009 distribution of PubMed (Björne *et al.*, 2010). The resulting dataset ('*PubMed'09*') contains 36.4M GGP symbols and 19.2M events pertaining to these entities. In subsequent work, the dataset was updated with citations from the period 2009–2011 (Van Landeghem *et al.*, 2012a), resulting in 40.3M tagged GGPs and 21.3M extracted events ('*PubMed'11*').

6.1.2 PubMed Central OA full-texts

Recently, we have extended the scope of EVEX from PubMed (PM) abstracts to additionally include full-text articles from the Open Access (OA) subset of PubMed Central. In the Shared Task of 2011, TEES achieved 54.37% F-score for abstracts, and 50.72% F-score for full text (Table 6.1). Performance is thus expected to be slightly lower on PMC full-text articles, but TEES is still within 2.5 pp. of the best performing system on full texts by Riedel and McCallum (2011). Further, it has been shown that BANNER maintains its high performance when moving from abstracts to full text (Cohen *et al.*, 2010). In general, processing full-text data is known to be more difficult, maybe partly due to the lower frequency of items of interest (Section 6.1.3).

From the PMC OA dataset, a set of 372K full-text articles was processed by first converting the XML format to ASCII text, building on software introduced for the ST'11 (Stenetorp *et al.*, 2011a). The GENIA Sentence Splitter (Kazama and Tsujii, 2003) further divides the text into sentences before BANNER and TEES are applied. As a result, the EVEX dataset of 21.3 million events previously extracted from PubMed abstracts

	Abstracts	Full text	Total
Articles	5.8M	0.3M	6.1M
Sentences	48.9M	54.3M	103.2M
GGPs	39.3M	27.9M	67.3M
Events	20.8M	13.5M	34.3M

Table 6.2: Number of extracted GGPs and biomolecular events in the EVEX'12 dataset, only showing statistics for documents with at least one identified GGP.

was augmented with an additional 13.5 million events. When merging these two datasets ('*EVEX'12*'), the PubMed abstracts corresponding to PMC articles were removed, preventing artificial data duplication. As a result, the cleaned PM abstract data contains 20.8 million events.

Processing all the 372K PMC full-text articles roughly took about 9429 CPU hours. To make the processing times manageable in practice, the pipeline was parallelized over a hundredfold on cluster machines, resulting in several days of actual runtime. This demonstrates that, with current computational resources, bibliome-wide text mining is computationally feasible.

6.1.3 Data statistics

From the 21M PM abstracts and 372K PMC full-texts, 5.8M abstracts and 313K full texts contained at least one sentence with a GGP symbol. The remaining articles were considered to be out-of-scope for biomolecular event extraction. Table 6.2 depicts the final number of articles, sentences, GGPs and biomolecular events in the EVEX database.

While there are only 313K relevant PMC full-text articles, in total they contain more sentences than the 5.8M PubMed abstracts. However, on average the full text contains only 50 GGPs and 20 events per 100 sentences, compared to 80 GGPs and 40 events in 100 sentences from abstracts. These numbers illustrate how full texts are more sparsely populated with biomolecular interaction data.

Analysing the added value of processing full-text articles rather than just abstracts, Figure 6.1 plots the percentage of events extracted from the body of the full text that were also found in the corresponding abstract, plotted as a function of the number of times the event was found in the body text[4]. When found only once in the body, 2.7% of events also appear in the abstract. While this percentage increases as the event is found more often in the full document, there is still less than 50% chance of extracting an event from the abstract when it is repeated in the full document 10 times, even though these events are expected to reflect information of substantial focus in the article.

Overall, only 7% of all events extracted from the body of a full-text PMC article could also be found in its abstract, showing the necessity of analysing the full text rather

[4]Events are considered equal when they have the same structure and pertain to the same gene IDs.

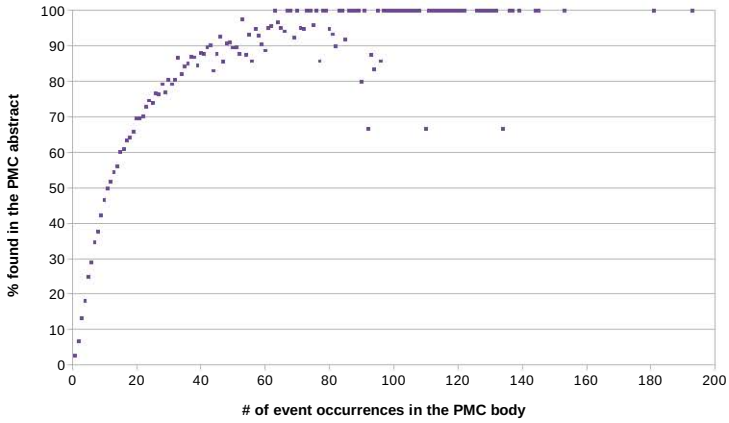


Figure 6.1: Analysis of the percentage of events found in the body of a PMC article that are also extracted from the corresponding abstract.

than just the abstract. When extending the search to all PMC abstracts, recall rises to 14%, and is further improved to 37% when incorporating all PubMed abstracts. While this analysis proves the usefulness of executing large-scale studies to improve on recall, at the same time it becomes clear that the full texts contain a wealth of information that can not be deduced by only processing abstracts.

6.1.4 Event ranking

To rank the extracted events according to their reliability, we have implemented an event scoring algorithm based on the output of the Turku Event Extraction System. Every classification is given a confidence score, the distance to the decision hyperplane of the linear classifier, where higher scores are associated with more confident decisions. There is not a single master classifier to predict the events in their entirety. Rather, individual classifications are made to predict the event trigger and each of its arguments. In order to assign a single confidence score to a specific event occurrence, the predictions from these two separate classifiers must be aggregated.

The confidence scores of the two different classifiers are not directly mutually comparable and we therefore first normalize all scores in the dataset to zero mean and unit standard deviation, separately for triggers and arguments. Subsequently, the score of a specific event occurrence is assigned to be the minimum of the normalized scores of its event trigger and its arguments, i.e. the lowest normalized confidence among all classification decisions involved in extracting that specific event. Using minimum as the aggregation function roughly corresponds to the fuzzy and operator in that it requires all

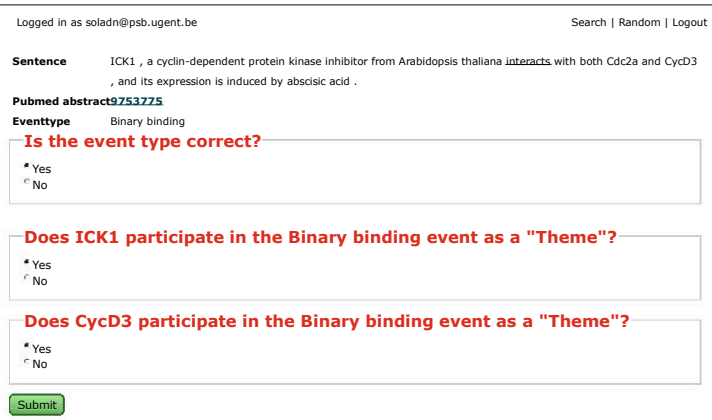


Figure 6.2: Screenshot of the evaluation website designed to perform the PLEV evaluation of TEES.

components of an event to be confident for it to be ranked high. Finally, the score of a generalized event is the average of the scores of all its occurrences.

To assign a meaningful interpretation to the normalized and aggregated confidence values, events within the top 20% of the confidence range are classified as 'Very high confidence'. The other 4 categories, each representing the next 20% of all events, are respectively labeled as 'High confidence', 'Average confidence', 'Low confidence' and 'Very low confidence'.

6.2 Event extraction performance

The official results of the Turku event extraction system in both the ST'09 and ST'11 are detailed in Table 6.1. The datasets used in the evaluations of these challenges are derived from GENIA, a corpus containing manual annotations for PubMed abstracts retrieved with the keywords 'human', 'blood cells' and 'transcription factors'. As a consequence, all event extraction systems are trained and evaluated on abstracts of this specific topic. In this section, an evaluation framework is designed to assess the cross-domain generalizability of the TEES classifier. Specifically, manual evaluations have been performed on a subset of the PubMed'09 dataset involving *Arabidopsis thaliana*, assessing the cross-species and cross-domain generalizability of the classifier.

6.2.1 Manual event evaluation

Figure 6.2 depicts a screenshot of the framework designed for manual evaluation of the event predictions. Event predictions made by any event extraction framework can be used as underlying data. In this chapter we focus on TEES, the best performing system

of the ST'09, and the EVEX dataset which is the result of running TEES on the whole of PubMed and PMC OA (Section 6.1).

For the evaluation, a set of 1176 PubMed abstracts were retrieved from CORNET (De Bodt *et al.*, 2010), a data integration platform for *Arabidopsis thaliana*. In these abstracts, 7691 events were extracted by TEES and thus copied to the internal database of the evaluation framework.

All manual evaluations in this framework have been performed by a Biotechnology master student, with no prior knowledge on text mining datasets, algorithms or evaluations. She was simply asked to judge whether a given event statement was actually expressed by the sentence it was extracted from.

The evaluation focused on binding and phosphorylation events, though some transcription and gene expression events were also evaluated. Events were selected randomly for each event type. In the second phase of the project, regulatory events were evaluated by selecting those on top of the evaluated physical events. If a recursive regulatory event contains a wrongly extracted physical event, the regulatory event is automatically annotated as being incorrect. Otherwise, it is presented for manual evaluation.

There are three types of regulatory events (pos/neg/unspecified). In this evaluation, the positive and negative regulatory events are also added to the unspecified category, simulating a use case where all regulatory events are presented to the user without specifying the type (Section 6.6).

6.2.2 Results

The resulting dataset, called, contains almost 1800 manually evaluated events (Table 6.3). Note that this evaluation setup can only judge precision of the results, as a recall assessment would require full annotation of all 1176 original abstracts.

Precision

Table 6.3 depicts the evaluation on the PLEV dataset and compares it to the official precision results of TEES in the ST'09. In general, the precision rates roughly match between the two evaluation datasets. There is however a notable drop in precision for phosphorylation and transcription events, comparing the new PLEV evaluation against the baseline ST'09 dataset. Further, regulatory events seem to perform better on the PLEV data. The gain in precision for unspecified regulations, from 38% on ST'09 to 62% on PLEV can partly be attributed to mixing the type of regulatory events as explained above. This result is in particular encouraging because many use cases involve finding regulators of certain GGPs, without a pre-defined requirement on the polarity of the regulation.

While there is thus some fluctuation of precision rates between the different event types, overall we conclude that the classifier, trained on a corpus involving human blood

Event type	PLEV		ST	
	pred.	prec.	pred.	prec.
Phosphorylation	314	64%	146	75%
Binding	695	49%	279	50%
Gene expression	53	81%	642	79%
Transcription	89	46%	78	69%
Protein catabolism	0	-	9	67%
Localization	0	-	105	82%
Negative regulation	52	65%	306	43%
Positive regulation	231	50%	782	49%
Regulation	418	62%	194	38%
All (Task 1)	1792	**58%**	2541	**58%**

Table 6.3: Precision rates of TEES as evaluated against the PLEV dataset and the ST'09 test set. The PLEV evaluation depicts the number of evaluated instances and their precision score. The ST'09 evaluation reflects the number of correct predictions made by TEES on the test set.

cell transcription factors, is perfectly capable of generalizing results to other organisms such as *Arabidopsis*.

Confidences

The EVEX dataset contains confidence values automatically derived from the output of the SVM classifiers in TEES (Section 6.1.4). To assess whether these confidence values actually correlate with the probability of an event being correct, they were plotted against the manual evaluations in PLEV. Figure 6.3 shows the resulting plot for phosphorylation events; similar results were obtained for the other event types.

The average precision of all phosphorylation events in PLEV is 64%. When a threshold is defined for the confidence value, and events below that threshold are disregarded, precision performance starts increasing. For example, only keeping events above the confidence threshold of -0.7 results in an average precision of 80%. The graph in Figure 6.3 further shows that the events with the highest confidence score, between 0.6 and 0.9, are all extracted correctly. These results validate the applicability of the confidence values to rank the text mining results in EVEX from highly reliable to less reliable predictions. Consequently, it becomes possible to only keep a subset of high-precision results when this is required for a specific use case.

6.3 Normalizing GGP symbols

Widely known biomolecular events occur in many different articles and their GGP arguments are often denoted with various synonyms and lexical variants. Canonicalization

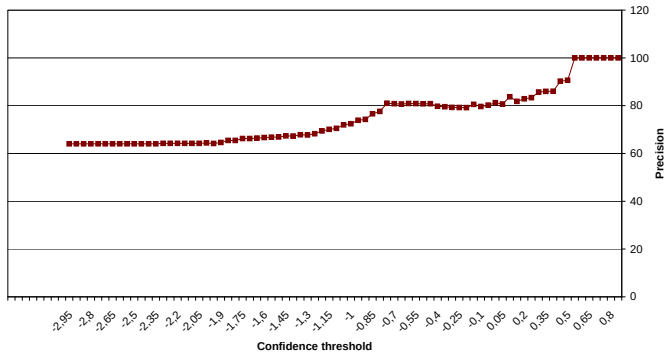


Figure 6.3: Precision of the phosphorylation events plotted against the automatically assigned confidence thresholds.

of the GGP symbols found in text deals with these lexical variants (Section 6.3.1), while the disambiguation algorithm uniquely links canonical forms to gene families (Section 6.3.2). As part of a recent update to the EVEX resource, we have further included full gene name normalization results, uniquely identifying the correct gene identifier for the ambiguous gene symbols from text (Section 6.3.3).

The analyses of the next first two sections have been performed on the PubMed'09 text mining data. Consequently, some of the tables contain slightly outdated information, but this is unavoidable as the resource is under constant development. However, the general findings and conclusions remain unchanged.

6.3.1 Canonicalization of entities

The GGP occurrences predicted by BANNER follow the guidelines of GENETAG, the corpus it was trained on. These guidelines not only allow gene and gene products, but also related entities such as protein complexes and gene promoters. Furthermore, BANNER frequently tags noun phrases such as *wild-type Esr-1 gene* rather than only the minimal symbol *Esr-1*. To enable integration of text mining results with external databases, it is necessary to refine the GGP mentions to canonical forms that can be linked to gene records such as those in Entrez Gene. To this end, common prefixes and suffixes such as 'gene' and 'wild-type' should be removed.

Finding generic affixes

In a first step towards canonicalization of the entities, a mapping table was assembled containing common contexts in which a gene symbol appears and where the full noun phrase can be reduced to that embedded symbol for the sake of information retrieval

GGP contexts
−ORG− −GGP− gene
−GGP− sequences
mutant −GGP− proteins
−GGP− homologs
cytoplasmic wild-type −GGP−

Table 6.4: This table lists a few examples of entity occurrences extracted with BANNER that are resolved to the embedded minimal gene symbol (marked as −GGP−).

(Table 6.4). This mapping table was created by matching[5] a list of likely minimal gene symbols to the extracted BANNER entities in two steps.

First, a list of likely minimal gene symbols is defined by looking for single token strings that were tagged by BANNER at least 50% of the times they occur in PubMed. Around 15,000 such likely gene symbols were found.

Subsequently, all multiple-token BANNER entity occurrences that contain such a likely minimal gene symbol are selected. The likely gene symbol is substituted with −GGP−, resulting in generalized contexts. These contexts are further generalized by replacing all known organisms with an −ORG− placeholder, using an extensive list of organism names from the Linneaus distribution (Gerner *et al.*, 2010) and a small collection of miscellaneous non-formal organism terms (e.g. *monkey*). Finally, all contexts are discarded where the embedded GGP is likely to be functionally too far removed from the embedding noun phrase (e.g. '−GGP− inhibitor'), relying on the corpus of entity relations (Chapter 5). Some of the contexts that were retained after this step, such as '−GGP− mutant' or '−GGP− promoter' still refer to entities that are distinctly different from the embedded GGP. However, these results are considered valid, as the goal of the affix stripping algorithm is to increase recall and offer explorative results involving various types of information on gene symbols.

The final list of contexts, generalized with −GGP− and −ORG− placeholders, is split into two separate lists of prefixes and suffixes, ranked by frequency. Numerical affixes as well as those shorter than 3 characters are discarded from these lists.

Recursively removing affixes

To canonicalize each text-bound GGP mention in the data, a gene symbol dictionary is assembled by extracting all gene names and symbols from Entrez Gene. As this list may contain common English words such as *was* and *protein*, only those were selected that were likely to be standalone gene symbols. This is calculated by $C_s/(C_s + C_n)$ where C_s is the number of times a string is tagged standalone and C_n is the number of times

[5] All string matching steps have been implemented using the *SimString* string retrieval library (Okazaki and Tsujii, 2010).

	p	r	F
No stripping	39.9	67.5	50.2
Affix stripping	48.7	82.3	61.1

Table 6.5: Influence on precision, recall and F-measure (given as P/R/F) of the affix stripping algorithm on the entity recognition module, as measured across all BioNLP'09 ST entity occurrences.

the string occurs in PubMed but is not tagged (neither as standalone, nor as part of a larger entity). This likelihood represents the proportion of standalone occurrences of the string that are tagged. We experimentally set a threshold on this value to be higher than 0.01, excluding a list of 2,865 common English words.

The algorithm for canonicalization then proceeds as follows:

1. Replace all organism names with the placeholder `-ORG-`
2. If the string can be matched[6] to a known symbol in Entrez Gene, stop the algorithm
3. Find all occurring affixes and strip the one associated with the highest count
4. Repeat (2-3) until no more affixes match
5. Strip remaining `-ORG-` placeholders, all whitespace and non-alphanumeric characters

For example, the canonicalization of *human anti-inflammatory il-10 gene* proceeds as
→ *-ORG- anti-inflamatory il-10 gene*
→ *anti-inflammatory il-10 gene*
→ *anti-inflammatory il-10*
→ *il-10*
at which point *il10* is matched in Entrez Gene, becoming the final canonical form.

Results

The affix stripping step of the canonicalization algorithm often substantially shortens the GGP symbols and an evaluation of its impact is thus necessary. One of the primary objectives of the canonicalization is to increase the proportion of extracted GGP mentions that can be matched to Entrez Gene identifiers. Its impact is evaluated using manually tagged entities from the ST'09 training set, which specifically aims at identifying mentions that are likely to match gene and protein symbol databases (Kim *et al.*, 2009).

We compare[7] the performance of the BANNER output before and after affix stripping (Table 6.5). The affix stripping results in a notable gain in both precision and recall.

[6] The comparison is done ignoring whitespace and non-alphanumeric characters.

[7] The comparison is performed on the level of bags of strings from each PubMed abstract, avoiding the complexity of aligning character offsets across different resources.

In particular, the nearly 15 pp. gain on recall clearly demonstrates that the affix stripping results in GGP strings more likely to match existing resources.

6.3.2 Family-based disambiguation

In this section, we describe how the canonical forms are assigned to unique gene families in an attempt to reduce symbol ambiguity.

Data collection

The first step towards gene symbol disambiguation involves collecting all possible gene symbols. From Entrez Gene (EG), 8M gene identifiers were retrieved linking to 10M unique symbols. All symbols are stripped of whitespace and non-alphanumeric characters to match the final step in the canonicalization algorithm.

Some of the gene symbols are highly ambiguous and uninformative, such as *NEWENTRY*. Others are ambiguous because they are abbreviations (Section 1.3.4). Finally, many symbols can not be linked to one unique gene, but do represent a homologous group of genes sharing a similar function. Often, orthologs with similar functions are assigned similar official gene names. In the next step, the EG gene symbols are thus resolved to gene families from HomoloGene or Ensembl.

The HomoloGene (HG) database is hosted at NCBI and provides the results of automated detection of orthologs in 20 completely sequenced eukaryotic genomes (Sayers *et al.*, 2010). From this resource, around 43,700 families were extracted, containing about 242,000 distinct genes. A second set of gene families was retrieved from Ensembl (ENS) (Flicek *et al.*, 2011). More than 13,000 families were assembled comprising about 220,000 vertebrate genes and 330,000 more families with over 2,446,000 genes are included from Ensembl Genomes, which provides coverage for metazoa, plants, protists, fungi, and bacteria (Kersey *et al.*, 2010).

As a general rule, the functional similarity scores per homologous pair in a gene family are higher when more stringent criteria are used to define the families (Hulsen *et al.*, 2006). While HomoloGene consists of many strict clusters containing true orthologs, bigger Ensembl families were obtained by assembling all pairwise orthologous mappings between genes. Ultimately, such clusters may also include paralogs, genes originated by duplication. As an example, consider the *nhr-35* gene from *C. elegans*, which has both *Esr-1* and *Esr-2* as known orthologs, resulting in the two paralogs being assigned to the same final Ensembl cluster. The Ensembl clustering algorithm thus provides a more coarse-grained method while the HomoloGene mapping results in more strictly defined gene families. The implications are discussed on a specific use case in Section 6.6.3.

Family	Type of symbol	Count
HG:47906	Default symbol	7
HG:99739	Synonym	1
HG:3740	Synonym	1
ENS:10415	Default symbol	12
ENS:8731	Synonym	1
ENS:8226	Synonym	1

Table 6.6: Intrinsic ambiguity of *esr1*, analysed in both HomoloGene and Ensembl families.

Disambiguation pipeline

First, the ambiguity for all gene symbols is synthesized by counting their occurrences in the gene families. Each such relation records whether the symbol is registered as an official or default gene symbol, as the gene description, an abbreviation, or a synonym. As an example, Table 6.6 depicts the intrinsic ambiguity of *esr1*.

In a subsequent step, the ambiguity is reduced by applying the following set of rules to each symbol, relying on a priority list imposed on the symbol type, ensuring an official or default name receives priority over a description or synonym.

1. If one family has the most (or all) hits for a certain symbol and these hits refer to a symbol type having priority over other possibilities, this family is uniquely assigned to that symbol.
2. If a conflict exists between one family having the highest linkage count for a certain symbol, and another family linking that symbol to a higher priority type, the latter is chosen.
3. If two families have equal counts and type priorities for a certain symbol, this symbol can not be unambiguously resolved and is removed from further processing.
4. If the symbol is not removed in the previous step but some ambiguity still remains, all families with only one hit for this symbol are removed, and steps 1-3 repeated.

The above disambiguation rules were applied to the 458K gene symbols in HomoloGene. In the third step, 6,891 symbols were deleted, and when the algorithm ends, 555 symbols remained ambiguous. In total, 451K gene symbols could thus be uniquely linked to a HomoloGene family (98%). In the *esr1* example depicted in Table 6.6, only the link to HG:47906 is retained. The results for Ensembl are very similar, with 342K out of 346K symbols uniquely resolved (99%).

Results

The symbol to gene family disambiguation algorithm successfully resolves almost all gene symbols in HomoloGene or Ensembl families. However, not all genes mentioned

	Distinct symbols		Occurrences	
Canonical	3235.0K	100.0%	67.3M	100.0%
HomoloGene	85.5K	2.6%	35.4M	52.6%
Ensembl	114.2K	3.5%	40.0M	59.5%
Ensembl Genomes	141.5K	4.4%	40.2M	59.8%

Table 6.7: GGP coverage comparison, showing the number of distinct canonical GGP symbols as well as the number of different occurrences covered, out of the total number of 67.3M extracted BANNER entities in the EVEX'12 data.

in text are a member of a known gene family, and the event generalization on top of the gene families thus inevitably discards a significant portion of the text mining results.

Table 6.7 shows that only a small fraction of all unique canonical GGPs matches the gene families from HomoloGene or Ensembl (Genomes) (between 2 and 5%). However, this small fraction of symbols accounts for more than half of all GGP mentions in the text mining data, with the exact percentage depending on the generalization (between 52 and 60%). The family disambiguation algorithm thus discards a long tail of very infrequent canonical symbols.

Finally, it is to be noted that the family-based disambiguation presented here always includes a few false positive hits, for example when events mention *Esr-1* as the (much less common) abbreviation for *Enhancer of shoot regeneration* and the canonical form is resolved to the family of Estrogen receptors. These false positives may be prevented by taking into account local context such as organism mentions, as the *Enhancer of shoot regeneration* gene is only present in *A. thaliana*. To resolve these issues, a full gene name normalization system is presented in the next section.

6.3.3 Gene normalization

Gene normalization is the task of identifying the real-world object that a GGP mention in text refers to, usually cast as associating text strings to database identifiers (Section 1.3.4). For assigning Entrez Gene identifiers to the GGPs in EVEX, the GenNorm system was applied, which was among the best performing systems in the Gene Normalization task of the BioCreative III Challenge, achieving first rank by several evaluation criteria (Lu *et al.*, 2011).

GenNorm is an integrative method for cross-species gene normalization (Wei and Kao, 2011) which covers the three major modules of the BioCreative III Gene Normalization task: gene name recognition (GNR), species assignment (SA) and species-specific gene normalization (SGN). In this study, we do not apply the GNR module, as the event pipeline already extracts GGP mentions with BANNER.

The first step in the normalization pipeline assigns a species to each GGP mention, using a dictionary-based matching method with two robust inferring strategies to gener-

Count		Scientific name	Most commonly used synonym
34.6M	51%	*Homo sapiens*	patients
9.9M	15%	*Mus musculus*	mice
5.8M	9%	*Rattus norvegicus*	rats
1.8M	3%	*Saccharomyces cerevisiae*	yeast
1.5M	2%	*Escherichia coli*	E. coli
0.9M	1%	*Drosophila melanogaster*	Drosophila
0.7M	1%	*Bos taurus*	bovine
0.7M	1%	*Arabidopsis thaliana*	Arabidopsis
0.6M	1%	*Human immunodeficiency virus 1*	HIV-1
0.5M	1%	*Oryctolagus cuniculus*	rabbit

Table 6.8: Top 10 most occurring organisms in EVEX, their most commonly used synonym in text and the number of assigned GGP occurrences.

ate a species lexicon. This lexicon combines organism and cell names to cover various species mentions and hints. Further, contextual information is used to deal with inter-species ambiguity of gene mentions. Finally, an inference network model is applied to resolve the intra-species gene ambiguity and variation, assigning unique EG identifiers to GGP mentions in text where possible.

The gene normalization results have only recently been added to EVEX, and they allow to improve upon the original family assignment algorithm as described in Section 6.3.2. In the new and improved version of this algorithm, Entrez Gene identifiers are first used to identify the correct family of gene mentions. When a specific gene mention in text could not be normalized by GenNorm, we resort to the previously introduced procedure that resolves the canonical form of a gene symbol to the most likely gene family. This two-step approach enhances precision due to the detailed normalization procedure while still maintaining high recall.

Results

The normalization algorithm assigns unique identifiers to 28.6M (43%) automatically extracted GGP mentions, identifying more than 120K different EG identifiers in total. The remaining 57% entities could not be disambiguated to a unique gene ID. This can be explained partly by the broad scope of BANNER, which tags not only GGP symbols but also gene families, protein complexes and other molecules. While these additional entities are valuable for information retrieval purposes, they are not expected to be normalized by GenNorm.

For the GGPs without unique gene ID, GenNorm still assigns the most plausible taxonomy ID. In total, more than 4800 different species are recognised across the whole dataset, ranging from viruses, bacteria and fungi to plants and animals (Table 6.8). These

annotations in the dataset allow for filtering of information on specific taxonomy identifiers, as well as specifically retrieving cross-species events such as the binding event expressed in the sentence '*Radiolabeled human beta 2-microglobulin can bind to mouse histocompatibility antigens on the cell surface*'.

6.4 Normalizing event structures

When presenting the dataset of text mining results to researchers not in BioNLP, the arbitrarily complex event structures may prevent intuitive understanding of the contained information. For this reason, several novel abstract layers are implemented on top of the existing data, providing data refinement (Section 6.4.1), a pairwise view (Section 6.4.2) and the generation of indirect associations (Section 6.4.3). These algorithms have been analysed on the PubMed'11 dataset.

In a final step, the refined event structures and normalized GGP symbols are used to generalize the text mining events to their homology-based variants (Section 6.4.4).

6.4.1 Event refinement

The extraction of event structures is highly dependent on the lexical and syntactic constructs used in the sentence, and may therefore contain unnecessary complexity. This is because the event extraction system is trained to closely follow the actual statements in the sentence and thus, for instance, marks both the words *increase* and *induces* as triggers for positive regulation events in the sentence *Ang II induces a rapid increase in MAPK activity*. Consequently, the final event structure is extracted as

```
Pos-Reg(C:Ang II, T:Pos-Reg(T:MAPK))
```

In other words, *Ang II* is a cause argument of a positive regulation event, which has another positive regulation event as its theme.

While correctly extracted, such nested single-argument regulatory events, often forming chains of several events long, add little additional information and it is desirable to simplify them before they are presented to the users of the EVEX dataset. Clearly, the event above can be restated as

```
Pos-Reg(C:Ang II, T:MAPK)
```

by removing the nested single-argument positive regulation event. This refinement helps to establish the event as equivalent with all other events that can be refined to the same elementary structure, enhancing the event aggregation possibilities in EVEX. However, when presenting the details of the extracted event to the user, the original structure of the event is preserved.

Table 6.9 lists the set of refinement rules. In this context, positive and negative regulation refer to having a general positive or negative effect, while an unspecified regulation

Original	Result	Example
Pos(C, T:Pos)	Pos(C, T)	BRs induce accumulation of BZR1 protein
Pos(C, T:Reg)	Pos(C, T)	PKS5 mediates PM H + - ATPase regulation
Reg(C, T:Pos)	Pos(C, T)	CaM regulates activation of HSFs
Neg(C, T:Neg)	Pos(C, T)	E2 prevented down-regulation of p21
Reg(C, T:Reg)	Reg(C, T)	PDK1 is involved in the regulation of S6K
Neg(C, T:Reg)	Neg(C, T)	GW5074 prevents this effect on ENT1 mRNA
Neg(C, T:Pos)	Neg(C, T)	BIN2 negatively regulates BZR1 accumulation
Reg(C, T:Neg)	Neg(C, T)	The effect of hCG in down-regulating ER beta
Pos(C, T:Neg)	Neg(C, T)	DtRE is required for repression of CAB2

Table 6.9: Listing of the refinement rules, involving any nested combination of the three types of regulation: positive regulation (Pos), negative regulation (Neg) and unspecified regulation (Reg). Each parent event has a regulatory (T)heme argument and an optional (C)ause. The nested regulations are all regulations without causes and their detailed structure is omitted for brevity. In full, the first structure would read Pos(C:A, T:Pos(T:B)) which is rewritten to Pos(C:A, T:B) with GGP_A and GGP_B being two GGPs.

could not be resolved to either category due to missing information in the sentence. The rules are repeatedly applied to each event, proceeding from top to bottom, as long as any rule matches.

To simplify the single-argument regulatory events, we proceed iteratively, removing intermediary single-argument regulatory events as long as any rule matches. A particular consideration is given to the polarity of the regulations. While a nested chain of single-argument positive regulations can be safely reduced to a single positive regulation, the outcome of reducing chains of single-argument regulations of mixed polarity is less obvious. As illustrated in Table 6.9, application of the rules may result in a change of polarity of the outer event. For instance, a regulation of a negative regulation is interpreted as a negative regulation, changing the polarity of the outer event from unspecified to negative. To avoid excessive inferences, the algorithm only allows one such change of polarity. Any subsequent removal of a nested single-argument regulatory event that would result in a type change, forces the new type of the outer event to be of the unspecified regulation type.

Results

By removing the chains of single-argument regulatory events, the refinement process simplifies and greatly reduces the heterogeneity in event structures, facilitating semantic interpretation and search for similar events. The process reduces the number of distinct event structures by more than 60%.

Further, the refinement algorithm increases the number of events with more than one gene symbol as a direct argument from 1471K to 1588K, successfully generating

more than a hundred thousand simplified events that can straightforwardly be parsed for pair-wise relations (Section 6.4.2).

However, it has to be noted that the results of the refinement algorithm are merely used as an abstract layer to group similar events together and to offer quick access to relevant information. The original event structures as extracted by TEES are always presented to the user when detailed information is requested, allowing the user to reject or accept the inferences made by the refinement algorithm.

6.4.2 Pairwise abstraction

The EVEX resource is centered around GGPs, i.e. genes, proteins and mRNA. The most important supported functionality is the identification and categorization of pairs of related GGPs, as this pairwise point of view comes natural in the life sciences. It can be implemented on top of the events with ease by analysing common event structures and defining argument pairs within. The refinements discussed in the previous section substantially decrease the number of unique event structures present in the data, restricting the required analysis to a relatively small number of event structures. Furthermore, only those events need to be considered that involve more than one GGP, or that are a recursive argument in such an event, reducing the set of 21M event occurrences in the PubMed'11 data to 12M.

As an example, consider the event

`Pos-Reg(C:`*Thrombin*`,T:Pos-Reg(C:`*EGF*`,Pho(T:`*Akt*`)))`

extracted from the sentence *Thrombin augmented EGF-stimulated Akt phosphorylation*. The pairs of interest here are (Thrombin,Akt) and (EGF,Akt), and both associations are coarsely categorized as *regulation*. Therefore, when searching for *Thrombin*, the *Akt* gene will be listed among the regulation targets and a search for *Akt* will list both *Thrombin* and *EGF* as regulators. Note, however, that the categorization of the association as *regulation* is only for the purpose of coarse grouping of the results. It is always possible to access the details of the original event.

There is a limited number of prevalent event structures which account for the vast majority of event occurrences. Table 6.10 lists the most common structures, together with the GGP pairs extracted from them. The algorithm to extract the GGP pairs from the event structures proceeds as follows:

1. All argument pairs are considered a candidate and classified as *binding* if both participants are themes of one binding event, and *regulation* otherwise[8].

2. If one of the GGPs is a Theme argument of an event which itself is a Cause argument, e.g. GGP_B in `*Reg(C:*Reg(C:`A`,T:`B`),T:`Z`)`, the association

[8]Note that due to the restrictions of event arguments, only binding and regulation events can have more than one argument.

Occ. %	Event pattern	GGP pair
58.6	Phy(T:A)	–
15.0	*Reg(T:A)	–
8.4	*Reg(T:Phy(T:A))	–
8.0	Binding(T:A,T:B)	A x B
4.7	*Reg(C:A,T:B)	A > B
3.8	*Reg(C:A,T:Phy(T:B))	A > B
0.2	*Reg(C:*Reg(T:Phy(T:A)),T:Phy(T:B))	A >> B
0.2	*Reg(C:Phy(T:A),T:B)	A >> B
0.2	*Reg(C:Phy(T:A),T:Phy(T:B))	A >> B

Table 6.10: The most prevalent event patterns in the (refined) EVEX data, considering only events with more than one GGP symbol, and their recursively nested events. These patterns refer to any type of regulation ($*Reg$), binding events of 2 GGPs, and any physical event (Phy). The left-most column refers to the proportion of occurrences covered by the given pattern and the right-most column depicts the extracted GGP pair and a coarse classification of its association type. A and B refer to GGP symbols and bindings are represented with `x`. Further, `A > B` means GGP_A *regulates* GGP_B while `A >> B` expresses an indirect regulation.

type of the candidate pair (GGP_B-GGP_Z) is reclassified as *indirect regulation*, since the direct regulator of GGP_Z is the cause argument of the nested regulation (GGP_A).

3. If one of the GGPs is a Cause argument of an event which itself is a Theme argument, e.g. GGP_A in *Reg(C:Z,T:*Reg(C:A,T:B)), the candidate pair (GGP_Z-GGP_A) is discarded.

While the association between G1 and G2 is discarded in step 3 since it in many cases cannot convincingly be classiffied as a regulation, it is represented as a *co-regulation* when indirect associations, described in the following section, are sought.

6.4.3 Indirect associations

A cell's activity is often organized into regulatory modules, i.e. sets of co-regulated genes that share a common function. Such modules can be found by automated analysis and clustering of genome-wide expression profiles (Segal *et al.*, 2003). Individual events, as defined by the BioNLP Shared Tasks, do not explicitly express such associations. However, indirect regulatory associations and functional modules can be identified by combining the information expressed in several distinct events. For instance, the events *Reg(C:A,T:Z) and *Reg(C:B,T:Z), can be aggregated to present the hypothesis that GGP_A and GGP_B co-regulate GGP_Z. Such hypothesis generation is much simplified by the fact that the events have been refined using the procedure described

Association	Interpretation
A > Z < B	A and B co-regulate Z
A < Z > B	A and B are being regulated by Z
A x Z x B	A and B share a common binding partner Z

Table 6.11: Indirect associations between GGP_A and GGP_B, established through a common interaction partner GGP_Z. Bindings are represented with x and for regulations A > B means that *GGP_A regulates GGP_A*.

in Section 6.4.1 and the usage of a relational database, which allows efficient querying across events (Section 6.5).

Currently, several indirect associations are implemented, precalculated and stored in the database, including co-regulation and common binding partners (Table 6.11). These links enable fast retrieval of e.g. co-regulators or GGPs that are targeted by a common regulator, facilitating the discovery of functional modules through text mining information. However, it needs to be stated that these associations are mainly hypothetical, as, for example, co-regulators additionally require co-expression. Details on gene expression events can be found by browsing the EVEX web application (Section 6.6).

6.4.4 Event generalizations

In order to gain a broader insight into the millions of extracted event occurrences, it is necessary to identify and aggregate multiple occurrences of the same underlying event. This generalization also notably simplifies working with the data, as the number of generalized events is an order of magnitude smaller than the number of event occurrences.

To aggregate event occurrences into generalized events, it is necessary to first define equivalence of event occurrences: two event occurrences are equivalent if they have the same event type, their event structure is equivalent, and their core arguments are equivalent and have the same semantic roles. Equivalence of event structures is determined after applying the refinement rules described in Section 6.4.1. For arguments that are themselves events, the equivalence is applied recursively. The equivalence of GGP arguments can be established in a number of different ways, affecting the granularity of the event generalization.

One approach is to use the string canonicalization described in Section 6.3.1; two GGP arguments are then equivalent if their canonical forms are equal. However, while this approach accounts for lexical variation, it does not take symbol synonymy into account. A different approach which we believe to be more powerful, is to disambiguate GGP symbols to gene families, as described in Section 6.3.2. In this latter approach, two GGPs are deemed equivalent if their canonical forms can be resolved to the same gene family. Consequently, two event occurrences are considered equivalent if they pertain to the same gene families. Finally, generalized events can be built on top of the Entrez

	Events	Occurrences	
Canonical	2953K	34.3M	100.0%
Entrez Gene	748K	15.8M	46.2%
HomoloGene	1006K	21.8M	63.5%
Ensembl	1042K	23.5M	68.5%
Ensembl Genomes	1001K	21.4M	62.5%

Table 6.12: Event coverage comparison, showing the number of refined generalized events as well as the number of different occurrences covered, out of the total number of 34.3M text-bound event occurrences in the EVEX'12 data.

Gene identifiers defined in Section 6.3.3.

As all these approaches have their merits and support different use cases, five distinct generalization procedures have been implemented: one on top of the canonical gene symbols, one on top of each gene family definition (HomoloGene, Ensembl and Ensembl Genomes) and one on top of the Entrez Gene normalization results.

Results

The Entrez Gene normalization-based event generalization accounts for 15.8M (46.2%) biomolecular events on top of 28.6M normalized GGPs. Roughly 12.9M of these events can be fully mapped into Ensembl Genomes, accounting for 38% of all event occurrences. By further resolving the remaining ambiguous canonical forms to their most likely family, an additional set of 8.5M event occurrences can be mapped to Ensembl Genomes, extending the total coverage to 21.4M event occurrences (62.5%). This analysis, which shows similar results in HomoloGene and Ensembl, demonstrates how the two-step disambiguation procedure significantly improves on recall by using the canonical form of a GGP symbol when no EG ID could be assigned.

Table 6.12 shows the final statistics on the number of generalized events and event occurrences. The generalizations results in a considerably smaller number of events, while the family-based ones still account for more than 60% of all event occurrences (between 62% and 69%). These findings are in line with the numbers previously presented for coverage on GGPs (Section 6.3.2).

As part of a recent study on the regulation of NADP(H) expression in *E. coli*, few manual evaluations were also conducted (Kaewphan *et al.*, 2012). For about 250 correctly extracted events, the correctness of the assignment of their arguments to Ensembl Genomes families was evaluated. It was found that 72% of event occurrences had both of their arguments assigned to the correct family.

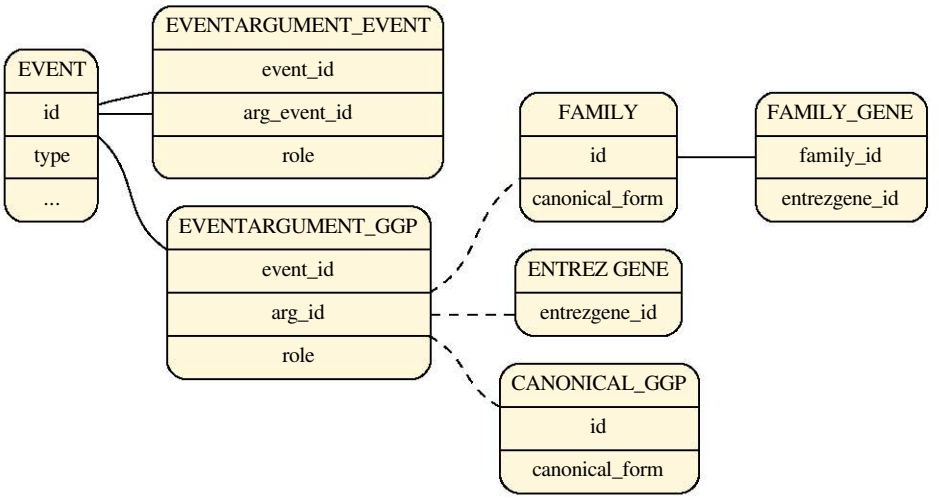


Figure 6.4: Database scheme of the generalized events. Of the general scheme (i.e. the three left-most tables), several instantiations exist in the database: canonical-based, EG normalization-based and family-based. The family-based generalization is further implemented using three distinct family definitions: HomoloGene, Ensembl and Ensembl Genomes. Following the dotted lines, each instance links to a different table in which the canonical forms and the gene identifiers can be retrieved (right-most tables).

6.5 MySQL database

As the original PubMed'09 events extracted by Björne *et al.* (2010) were purely text-bound and distributed as text files, they could not easily be searched. One important contribution of our follow-up work is the release of all text mining results as a MySQL database (Van Landeghem *et al.*, 2011b). During the conversion, all original information is kept, including links to the PubMed IDs and the offsets in text for all GGPs and triggers, referring to the original strings as they were obtained by BANNER and the event extraction system. Further, the new text mining events of the PubMed'11 update and from the PMC OA articles were added to the DB. This allows for fast retrieval of text mining data on a bibliome-scale.

As described in Section 6.4.4, five distinct generalization methods have been applied to the original events. On the database level, each generalization is represented by a separate set of tables for the generalized events and their arguments, aggregating important event statistics such as occurrence count and negation/speculation information (Figure 6.4). Finally, a mapping table is provided that links the generalized events to the event occurrences from which they were abstracted. Figure 6.5 shows the general scheme of the entire database.

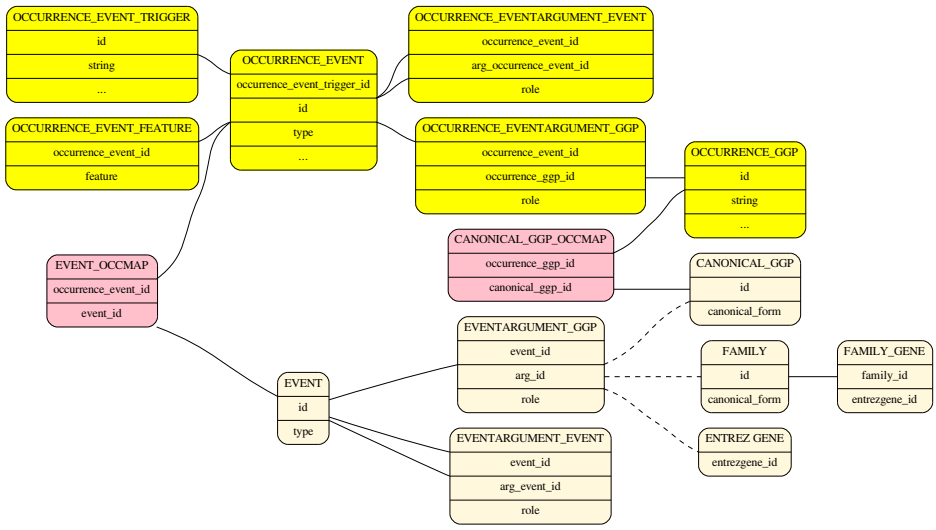


Figure 6.5: Overview of the database scheme. Both the text-bound occurrences (upper part) as well as the generalized part of the DB (lower part) follow a similar scheme. The original event and GGP occurrences are mapped to either the canonical generalization, the EG-generalization or one of the family-based generalizations.

The main target audience of the EVEX database is the BioNLP community (event occurrences) and bioinformaticians (generalized events). However, the dataset is not easily accessible for other researchers in the life sciences who are not familiar with the intricacies of the event representation. For this reason, we have additionally created a publicly available web application based on the EVEX dataset, bringing detailed text mining results closer to a broader audience of end-users including biologists, geneticists and other researchers in the life sciences. This web application and an example use case are described in the next section.

6.6 Web application

The primary purpose of the application is to provide the EVEX dataset with an intuitive interface that allows for explorative browsing of text mining results while not requiring familiarity with the underlying event representation. The application presents a comprehensive and thoroughly interlinked overview of all events for a given GGP or GGP pair.

This web interface is not the first text mining tool applied to a large scale. For instance, the *iHOP* (Hoffmann and Valencia, 2004) and *Medie* (Ohta *et al.*, 2006) sys-

tems allow users to directly mine literature relevant to given genes/proteins of interest, allowing for structured queries far beyond the usual keyword search. *EBIMed* (Rebholz-Schuhmann *et al.*, 2007) offers a broad scope by including also Gene Ontology terms such as biological processes, as well as drugs and species names. Other systems, such as the *BioText search engine* (Hearst *et al.*, 2007) and *Yale Image Finder* (Xu *et al.*, 2008) allow for a comprehensive search in full-text articles, including also figures and tables. Finally, the *BioNOT* system (Agarwal *et al.*, 2011) focuses specifically on extracting negative evidence from scientific articles.

The main difference between the EVEX application and other available large-scale text mining applications is that EVEX covers highly detailed event structures that are enriched with homology-based information, and additionally extracts indirect associations by applying cross-document aggregation of events. To illustrate the functionality and features of the web application, a use case is presented on a specific budding yeast gene, *Mec1*, which is conserved in *S. pombe*, *S. cerevisiae*, *K. lactis*, *E. gossypii*, *M. grisea* and *N. crassa*. *Mec1* is required for meiosis and plays a critical role in the maintenance of genome stability. Furthermore, it is considered to be a homolog of the mammalian *ATR/ATM*, a signal transduction protein (Carballo and Cha, 2007).

6.6.1 Finding direct and indirect associations

The main functionality of the EVEX resource is providing fast access to relevant information and related biomolecular entities of a GGP or pair of GGPs[9]. The most straightforward way to achieve this is by specifying an Entrez Gene, UniProt or GenBank ID, or by searching for a gene symbol on the canonical generalization (Section 6.3.1). The web application further allows for taxonomic filtering.

The result page generates a list of biomolecular events relevant to the query GGP, grouped by event type and ranked by confidence, ranging from (very) high to average and (very) low (Section 6.1.4). At the top of the page, an overview of all regulators, regulated genes and binding partners is provided, each accompanied with an example sentence. This coarse grouping is made possible by the pairwise abstraction described in Section 6.4.2. Further, co-regulators are listed together with the number of co-regulated genes (Section 6.4.3). Figure 6.6 shows the results when searching for *Mec1*. This overview lists 21 regulation targets, 11 regulators, 27 binding partners and 263 co-regulators.

Selecting the target *RAD9*, the web application visualises all event structures expressing regulation of *RAD9* by *Mec1* (Figure 6.7). This enables a quick overview of the mechanisms through which the regulation is established, which can have a certain polarity (positive/negative) and may involve physical events such as phosphorylation or protein-DNA binding. The different types of event structures are coarsely grouped into

[9] Analysis of large gene lists is currently not supported, as such a bioinformatics use case is already covered by the publicly available MySQL database.

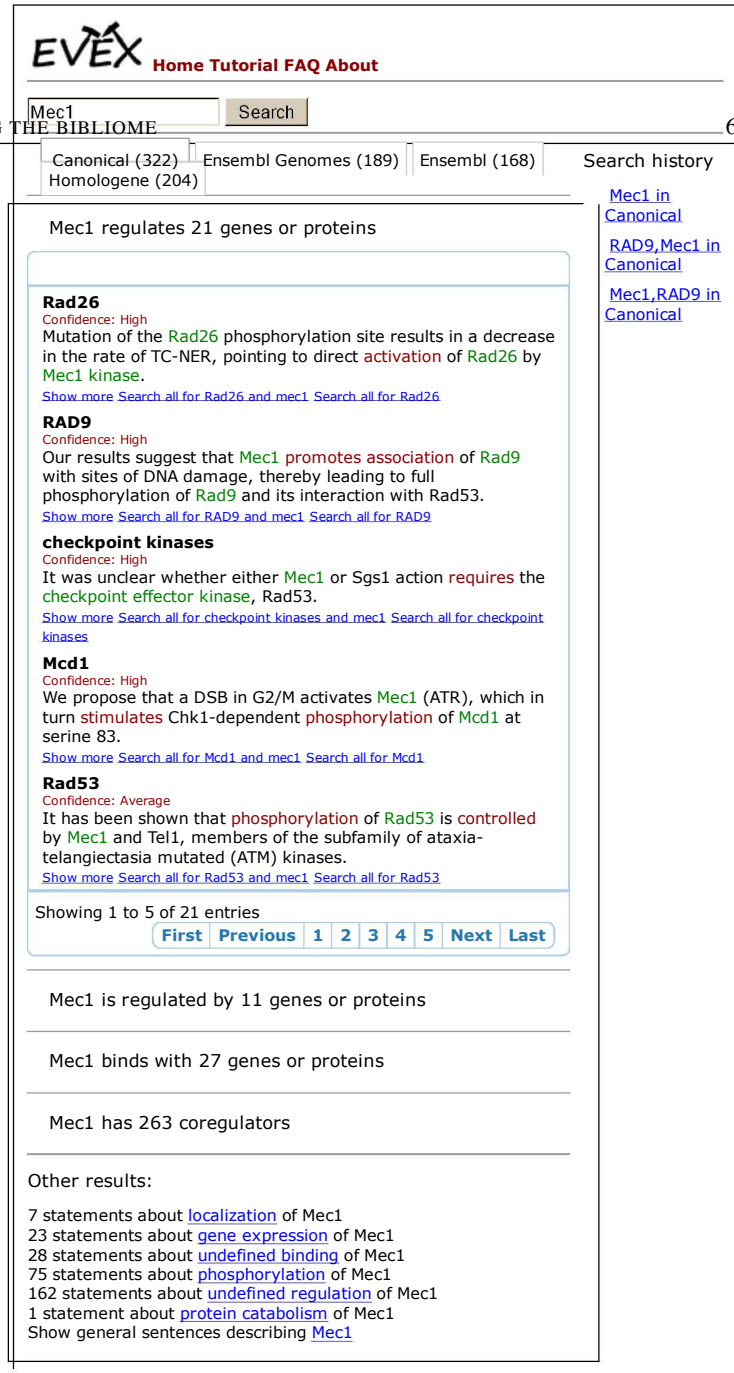


Figure 6.6: Search results for *Mec1* on the canonical generalization. An overview of directly associated genes is presented, grouped by event type, as well as relevant links to additional sentences and articles. Some events might contain speculative information.

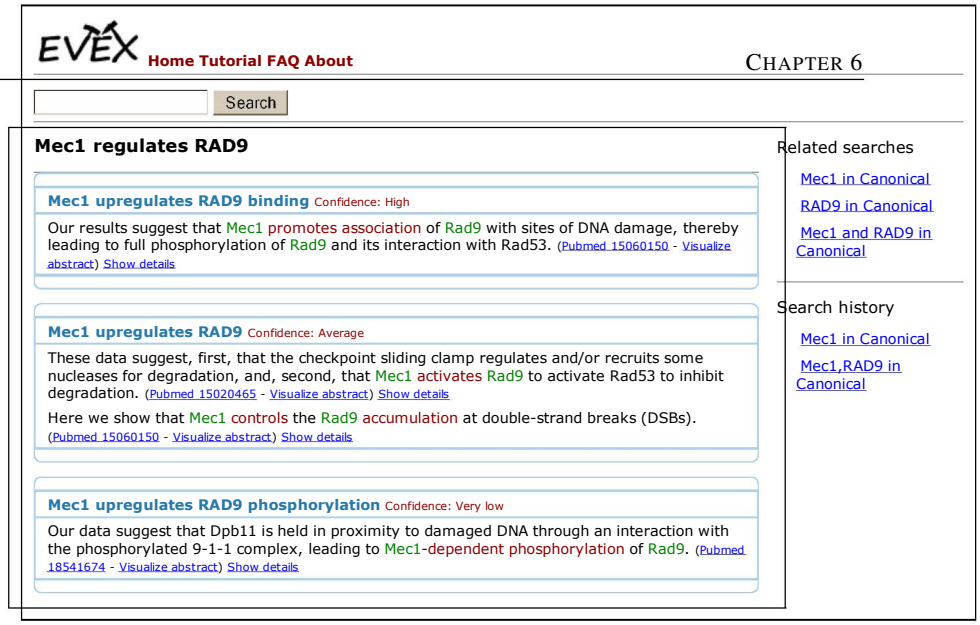


Figure 6.7: Detailed representation of all evidence supporting the regulation of *RAD9* by *Mec1*. Regulation mechanisms can have a certain polarity (positive/negative) and may involve physical events such as phosphorylation or protein-DNA binding.

categories of similar events, and presented from most to least reliable using the confidence scores.

Apart from the regulatory and binding mechanisms, the overview page of the GGP pair also provides conclusive evidence for a binding event between *RAD9* and *Mec1*. Further, potential co-regulations are listed, enumerating targets that are regulated by both genes, such as *Rad53*. When accessing the details of these results, all evidence excerpts supporting both regulations are presented. Other indirect associations, such as common regulators and binding partners, can be retrieved equally fast.

6.6.2 Retrieving sentences by event type

The overview page of *Mec1* (Figure 6.6) contains additional relevant information including links to sentences stating events of *Mec1* without a second argument, grouped by event type. While these events incorporate only a single GGP and may not be very informative by themselves, they are highly relevant for information retrieval purposes, presenting relevant sentences and articles describing specific processes involving the GGP of interest.

At the bottom of the overview page, a similar and even more general set of sentences can be found, providing pointers to relevant literature which still requires manual analysis. Such sentences, even though they contain no extracted events, may include useful

background information on the GGP such as relevant experimental studies, related diseases or general functions and pathways.

6.6.3 Homology-based inference

The EVEX resource builds upon the previously described family generalizations (Section 6.4.4) and can thus provide a summary of all events pertaining to a certain family when searching for one of its members.

For example, instead of only looking at the information for one particular GGP as described previously, the search can be extended through Ensembl Genomes, retrieving information on homologous genes and their synonyms. The generated listings of regulators and binding partners are structured in exactly the same way as before, but this time each symbol refers to a whole gene family rather than just one GGP.

Conducting such a generalized search for *Mec1*, EVEX retrieves interaction information for *Mec1* and its homologs. The resulting page presents not only results for the symbol *Mec1*, but also for common symbols which are considered synonyms on the gene-family level, such as *ATR*. This type of synonym expansion goes well beyond a simple keyword query. For each gene family present in the text mining data, a family profile lists all genes and synonyms for a specific family, linking to the authoritative resources such as Entrez Gene and the Taxonomy database at NCBI.

The EVEX resource includes several distinct methods of defining gene families (Section 6.4.4), each accommodating for specific organisms and use cases. For example, Ensembl Genomes uses rather coarse grained families resulting in a family of 19 conserved genes, including the budding yeast gene *Mec1*, its mammalian *ATR* orthologs and genes from green algea and *Arabidopsis*. In contrast, the corresponding HomoloGene family only includes the 6 conserved *Mec1* genes in the *Ascomycota*.

6.6.4 Manual inspection of text mining results

An important aspect of the EVEX web application is the ability to retrieve the original sentences and articles for all claims extracted from literature. In the previous sections, we have described how EVEX can assist in the retrieval of directly and indirectly associated GGPs. However, to be applicable in real-life use cases and to be valuable to a domain expert, it is necessary to distinguish trustworthy predictions from unreliable hypotheses. For this reason, the automatically derived confidence values are displayed for each extracted interaction. On top of those, the site provides the opportunity to inspect the textual evidence in detail.

Consider for example the phosphorylation of *RAD9*, regulated by *Mec1* (Figure 6.7). To allow for a detailed inspection of this event, the web application integrates the *stav* visualiser (Stenetorp *et al.*, 2011b), developed as a supporting resource for the ST'11 (Kim *et al.*, 2011). This open-source tool provides a detailed and easily graspable presenta-

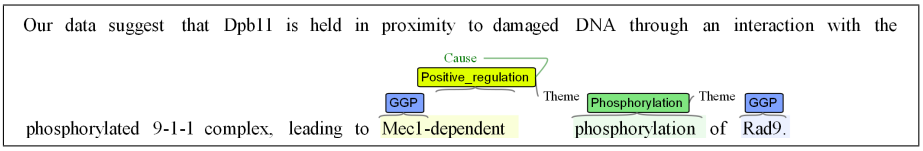


Figure 6.8: Visualization of a specific event occurrence by the stav text annotation visualiser. GGPs and trigger words are marked and connected to form events. Finally, arrows denote the roles of each argument in the event (e.g. Theme or Cause). This visualization corresponds to the formal bracketed format of the event: `Pos-reg(C: Mec1, T: Pho(T: RAD9))`.

tion of the event structures and the associated textual spans (Figure 6.8). To any user interested in the text mining details, this visualization provides valuable insights into the automated event extraction process. Additionally, the web application provides the opportunity to visualise whole PubMed abstracts with the stav visualiser, allowing a fast overview of event information contained within an abstract.

6.6.5 Site navigation

To easily trace back previously found results, a session-based search history at the right-hand side of the screen provides links to the latest searches issued on the site. Further, a box with related searches suggests relevant queries related to the current page. Finally, the web application provides a powerful method to browse indirectly associated information, by allowing the retrieval of nested and parent interactions of a specific event. For example, when accessing the details of *Mec1*'s regulation of *RAD9* phosphorylation and selecting the phosphorylation event, evidence is shown for many parent events involving different regulation polarities and various genes causing this specific phosphorylation. As such, we quickly learn that *RAD9* phosphorylation has many different potential regulators, such as *Ad5*, *Ad12* and *C-Abl*. This sort of explorative information retrieval and cross-article discovery is exactly the type of usage aimed at by the EVEX resource.

6.7 Conclusion

We have presented EVEX, a bibliome-wide text mining resource covering all PubMed abstracts and PMC OA full texts. EVEX contains more than 34 million biomolecular events extracted among 67 million GGP mentions from over 4800 species, ranging from viruses, bacteria and fungi to plants, animals and human. We will regularly update this resource to include new results from the latest publications.

The identified GGPs are canonicalized by stripping superfluous affixes (prefixes and suffixes) to obtain the core GGP symbol. The purpose of this canonicalization is to

abstract away from minor spelling variants and to deal with the fact that the BANNER named entity recognizer often includes a wider context around the core GGP symbol. The canonicalization algorithm itself cannot, however, deal with the ambiguity prevalent among the symbols. EVEX thus further resolves these canonicalized GGP symbols, whenever possible, into their most likely families. Finally, a recent addition of EVEX includes full gene normalization, assigning unique Entrez Gene IDs to 42% of all GGP occurrences. As such, this study presents the first bibliome-wide text mining analysis that combines complex event extraction with gene normalization, two major lines of research in the BioNLP community. The text mining systems applied here represent the state-of-the-art, as evaluated in community-wide shared task evaluations.

All relevant data are made publicly available as records in a MySQL database, including all original PM/PMC sentences, the extracted events and GGPs as well as the assigned canonical forms, gene families and Entrez Gene identifiers[10].

Further, a publicly available web application[11] has been developed that allows manual explorative browsing for supporting research in the life sciences. This application provides efficient and intuitive access to the large-scale event dataset by refining the complex event structures. Further, equivalent events are aggregated across articles, accounting for lexical variation and synonymy. This aggregation allows retrieving and summarizing relevant information across articles and species. Finally, the EVEX interface groups events with respect to the involvement of pairs of genes, providing the users with a familiar gene-centric point of view, without sacrificing the expressiveness of the event structures. This interpretation is extended also to combinations of events, identifying indirect associations such as common co-regulators and common binding partners, as a form of literature-based hypothesis generation.

We believe this resource to be very valuable for explorative analysis of text mining results and homology-based hypothesis generation, as well as for supporting future research on data integration and biomedical text mining.

[10] `http://bionlp.utu.fi/pubmedevents.html`
[11] `http://www.evexdb.org`

7 Discussion and future prospects

In this chapter, we summarize the contributions made to the BioNLP field in terms of novel algorithms, careful evaluations and critical assessments. Further, we discuss future prospects by illustrating several promising applications of the EVEX text mining dataset.

7.1 Information extraction

Among the most heavily studied tasks in BioNLP is the extraction of information about known associations between genes and gene products (GGPs). In this thesis, we have presented a novel machine learning framework (Chapter 2) that was applied to a wide range of extraction targets, including protein-protein interactions (Chapter 3), various biomolecular events including physical and regulatory interactions (Chapter 4) and non-causal relations between GGPs and domain terms (Chapter 5).

Further, we have described a bibliome-wide study on event extraction, producing, refining and generalizing biomolecular events for all PubMed abstracts and PMC Open Access full-text articles (Chapter 6). In this study, we gave an insight into the added value of processing full-text articles, as opposed to PubMed abstracts only. Analysis on the resulting EVEX resource has shown that a mere 7% of all events found in full text could also be extracted from the corresponding abstract, while 37% could be extracted from any abstract. These results underline the importance of extending the Open Access set of PubMed Central and the scope of existing text mining techniques.

Recently, the suitability of the EVEX dataset and web application to the task of

pathway curation was analyzed with a particular focus on recall (Ohta *et al.*, 2011). When analysing three high-quality pathway models, *TLR*, *mTOR* and yeast cell cycle, 60% of all interactions could be retrieved from EVEX. A thorough manual evaluation further suggested that, surprisingly, the most common reason for a pathway interaction not being extracted is not a failure of the event extraction pipeline, but rather a lack of semantic coverage. In these cases, the interaction corresponds to a relation type not defined in the ST'09 task and thus out of scope for the event extraction system. In the next few subsections, we describe possible future additions to EVEX to improve its semantic coverage.

7.1.1 Entity relations

Entity relations broaden the scope of text mining tools (Chapter 5): they function as hubs between events concerning similar molecular entities, improve on the level of detail provided by event extraction (Section 5.2) and finally they are useful for normalizing automatically extracted GGP symbols such as '*Esr-1* inhibitor' (Section 6.3.1 and 6.3.3). We have shown that we can predict the class of embedded entity relations, necessary for such normalization efforts, extremely well (Section 5.4.2). In future work, we aim at annotating semantic relations between molecular entities in the entire scientific literature, exploiting these relations for further refinements and improvements of EVEX.

7.1.2 Coreference resolution

All studies described in this thesis aim at extracting biomolecular information on a sentence-level, disregarding interactions expressed across different sentences. A particularly interesting topic for future work is combining coreference or anaphora resolution with dependency graphs in order to process events which span multiple sentences in text. Coreference resolution aims at finding various references to the same object mentioned in text, semantically linking phrases such as 'this gene' to a previously identified gene symbol. While cross-sentence text mining extraction is not supported by most PPI datasets (Section 3.1.1), there is some training data available in the Shared Task event corpora (Section 4.1.4), and we believe adding coreference resolution would be in particular useful when processing full-text articles.

7.1.3 Epigenetics

Further future work will focus on broadening the coverage of EVEX in terms of event types, particularly in the important domain of post-translational modifications and epigenetics. The system applied for the creation of EVEX obtained the highest performance for the 'EPI' subtask of the BioNLP ST'11 (Björne and Salakoski, 2011; Kim *et al.*, 2011). As the EPI events are structurally similar to the general events currently in EVEX, no major changes will be required in the underlying database schema.

7.2 Evaluations

Throughout the studies presented in this thesis, a key aspect of the analyses involves thorough evaluations and assessment of applicability of the novel algorithms to real-world use cases.

7.2.1 System-wide evaluations

For the extraction of PPIs, a lack of standard evaluation frameworks hinders adequate comparison of different methods. We have discussed these issues at length by evaluating the influences of the evaluation parameters on the final performance and proposing standard guidelines for the evaluation of PPI extraction systems (Section 3.1). Further, cross-dataset experiments were conducted to assess the cross-domain generalizability of PPI methods (Section 3.3.2). A similar experiment was performed on the core event predictions in EVEX, testing the generalizability of the classifier trained on human data, to a plant subset of abstracts we have evaluated using an in-house framework (Section 6.2).

Two different event extraction systems are discussed in this thesis. The system described in Chapter 2 and further used throughout Chapters 3, Chapters 4 and 5 was developed at Ghent University and achieved 5^{th} place out of 24 international teams participating in the official BioNLP'09 Shared Task (Section 4.3.2). Further, it ranked 2^{nd} for the REL sub-challenge of the ST'11 (Section 5.4.1). The system used to create the EVEX resource on the other hand (Section 6.1), was developed by the University of Turku, Finland, and was the winning system in the ST'09 and the ST'11 REL challenge.

Community-wide challenges such as the BioNLP Shared Task are extremely valuable to measure state-of-the-art performance and progress of various extraction systems. The software component GenNorm, used for gene name normalization in EVEX (Section 6.3.3) was among the best performing systems of another important community-wide challenge, BioCreative. By bringing together the best systems from the lines of research represented by the BioNLP ST and BioCreative challenges and applying them to the entire publicly available literature, we have created a text mining dataset of an unprecedented scope and level of detail, opening many new opportunities for the application of text mining data in integrative frameworks and applications in experimental studies, systems biology, database curation and comparative genomics (Section 7.3).

7.2.2 Parameter assessments

Instead of only benchmarking complete extraction systems on a specific corpus, we have additionally performed detailed analyses of various components and design choices of our extraction framework, such as different parsers, kernels and feature sets (Section 4.2). The analysis of various design choices is not only highly relevant for machine learning approaches, but can also offer a meaningful contribution to the development

of rule-based systems. The results of the feature selection experiments are particularly useful for this purpose.

First, we have performed FS experiments using the *Gain ratio* filter method for the PPI classification challenge (Section 3.2.3). A more advanced method was introduced for event extraction, using an ensemble of weak feature selectors (Section 4.4.1). This method further allowed us to gain a better insight into the specific challenges of event extraction (Section 4.4.5) and entity relation detection (Section 5.3.2).

These feature analyses have given us an in-depth understanding of the feature generation algorithms and ideas on how to improve on these. In particular, improvements to the trigger detection algorithm would allow reducing the number of candidate events and false positives. Another shortcoming that should be addressed in the future is the use of stemming. Stemming, though widely used, essentially just removes suffixes, preventing the algorithm to find equality between e.g. the stems 'present' and 'presenc'. A dictionary lookup to identify synonyms and related terms, could further reduce the sparseness of the feature vectors and create more general feature patterns. A final improvement for the lexical features could be the inclusion of N-grams for $N > 3$, as the feature clouds indicate that such features could be relevant for classification.

7.3 Applications

Among the typical use cases for BioNLP applications are support for content visualization, pathway and database curation, and hypothesis generation. Even though the semantic coverage of EVEX could still be improved upon, a few promising results were already obtained that illustrate interesting future directions. We describe these applications in the next few sections.

7.3.1 Explorative browsing

We have presented an intuitive web application that enables knowledge summarization and explorative browsing of text mining results (Section 6.6). In future work, we aim at extending this web application in particular through advanced search methods, personalization of results and support for manual curation of the automatically generated predictions. This creates the opportunity to discover meaningful relations through the cooperation of fully automated, supervised learning techniques on one hand, and an expert user able to interpret its results on the other hand.

The stable feature selectors presented in Section 4.4.1 could further guide the end-user through the results of automatic discovery by highlighting discriminative features used during classification. For instance, Figure 7.1 depicts a text sample highlighting top ranked features. These lexical constructs provide interesting clues about predicted events and help the reader to better understand the nature of the predictions made by the SVM classifier.

By electrophoretic mobility shift assays, this increase in **mRNA** was associated with a 5- to 10-fold increase in the STAT1-containing **DNA**-**binding complex** that **binds** to Fc gammaRI **promoter** elements.

Furthermore, the **tyrosine phosphorylation** of STAT1 and the **tyrosine kinases JAK1** and JAK2 was enhanced significantly in RGD-adherent monocytes compared with control cells.

Figure 7.1: Text sample from PMID:9278334. Three distinct event types are discussed: transcription (green, previous sentence), binding (purple, first sentence) and phosphorylation (red, second sentence). The relevant trigger words are 'binding complex' and 'phosphorylation' (underlined). Relevant BOW features include 'mRNA', 'DNA', 'binds', 'promoter' and 'tyrosine'. Finally, there is a match with the trigram 'tyrosin kinas protx'. All highlighted words help the reader find relevant clues for each event type.

7.3.2 Homology-based knowledge discovery

Functional annotation of genomes often requires extensive *in vivo* experiments. This time-consuming procedure can be expedited by integrating knowledge from closely related species (Fulton *et al.*, 2002; Proost *et al.*, 2009). Over the past few years, homology-based functional annotation has become a widely used technique in bioinformatics (Loewenstein *et al.*, 2009).

We have presented a similar functionality in EVEX, retrieving text mining results linked to gene families from HomoloGene and Ensembl (Section 6.6.3). This allows for a number of novel use cases such as retrieving relevant text mining events for newly discovered sequences.

7.3.3 Database curation

Through the gene normalization module, text mining results can be connected to the wealth of existing bioinformatics resources. To demonstrate, we analyse the EVEX database with respect to data from STRING, a rich resource of normalized protein associations incorporating data from many major domain databases, including high-throughput experiments, computationally inferred annotations and manually curated pathways (Jensen *et al.*, 2009).

First, all high-confidence protein pairs[1] are extracted that are supported by at least one direct source. The STRING DB internal protein identifiers of these pairs are then mapped to Entrez Gene identifiers, providing a set of 145K unique unordered protein

[1] Using the score > 0.7 threshold suggested in STRING documentation.

Source	Match	Total	Recall
PID	820	998	82.16
HPRD	694	1057	65.66
DIP	1738	4085	42.55
GRID	8346	28735	29.04
KEGG	19739	72620	27.18
MINT	2851	13805	20.65
IntAct	1984	10281	19.30
Reactome	1402	7871	17.81
BIND	1135	6453	17.59
BioCyc	25	810	3.09
PDB	717	32951	2.18

Table 7.1: Number of unique high-confidence protein pairs retrieved through STRING and supported by EVEX.

pairs.

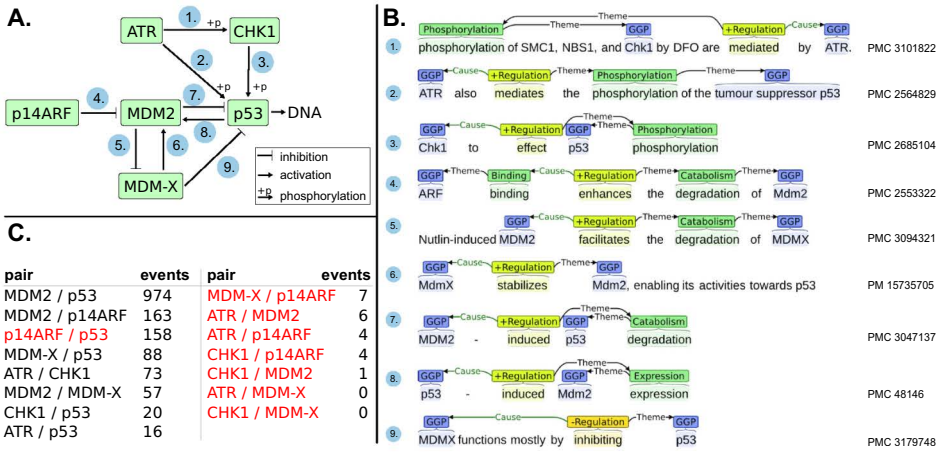
Subsequently, each pair of Entrez Gene IDs is searched in EVEX to determine whether events involving the two proteins have been extracted from the literature. Table 7.1 shows the results of this analysis, broken down by the source DB. There is a very broad variation in coverage, ranging from over 80% for the PID database, to just a few percent for BioCyc and PDB. Such a broad variation is to be expected: the PID database includes manually curated associations and requires literature support, while, for example, BioCyc includes sequence-based, computationally predicted associations, which are not expected to substantially overlap with existing literature.

The high recall against curated databases like PID illustrates the potential for text mining systems in database curation support, since the majority of the associations of interest are present in EVEX and accessible via the web interface. At the same time, this analysis demonstrates the necessity for data integration in the field of life sciences as the scope, reliability and coverage of different data resources vary significantly due to variation in initial design, aim and inference methods.

Additionally, negation information extracted from text (Section 4.2.7) might be used to locate inconsistencies between research articles and database facts, while speculative information (Section 4.2.8) could influence the confidence score of such database records. We regard these opportunities as interesting future work.

7.3.4 Pathway curation

One of the potential applications of event extraction is assisting pathway curation and analysis (Ohta *et al.*, 2011), a challenging discipline (Ghosh *et al.*, 2011). Previous exploratory work on connecting events to KEGG pathways has been limited to somewhat inaccurately matching proteins by name (Björne *et al.*, 2010). The recent addition of



C.

pair	events	pair	events
MDM2 / p53	974	MDM-X / p14ARF	7
MDM2 / p14ARF	163	ATR / MDM2	6
p14ARF / p53	158	ATR / p14ARF	4
MDM-X / p53	88	CHK1 / p14ARF	4
ATR / CHK1	73	CHK1 / MDM2	1
MDM2 / MDM-X	57	ATR / MDM-X	0
CHK1 / p53	20	CHK1 / MDM-X	0
ATR / p53	16		

Figure 7.2: Normalized events can be mapped to biomolecular pathways. A) Interactions of *p53* from KEGG pathway hsa04115. B) The highest confidence predicted event from EVEX for each **directed** KEGG interaction, linked through the assigned Entrez Gene IDs. All are correct and correspond to the KEGG interaction type. C) The number of events in EVEX for each **undirected** protein pair. Pairs not corresponding to a direct molecular interaction in *A)* are shown in red.

gene normalization data to EVEX (Section 6.3.3) now allows direct text mining support for pathway curation. To illustrate, Figure 7.2 (A) shows a subsection of a well-known KEGG pathway, the *human p53 signaling pathway* ('hsa04115' (Kanehisa and Goto, 2000)), which we attempt to recreate using EVEX.

For the reconstruction, each directed pair of proteins in the network is searched in EVEX, using the Entrez Gene IDs annotated in the KEGG pathway. Note that since these gene IDs refer to human genes, EVEX successfully restricts the result set to human biology. The number of extracted events for each protein pair is shown in Figure 7.2 (C). While most events correspond to direct physical interactions, statements in literature can also refer to indirect interactions, and this could be a source for many of the events on e.g. *p14ARF* interactions with *p53*.

For each directed KEGG interaction, the corresponding event with the highest confidence score is taken from EVEX, i.e. the event presented first to the users of the EVEX web interface (Section 6.6). These events are shown in Figure 7.2 (B).

KEGG interaction types are somewhat different from the event format, for example a KEGG *phosphorylation* event connecting proteins A and B would in the event scheme be annotated as

`Positive-Regulation(C:` A`, T: Phosphorylation(T:` B`)).`

Inspecting the example sentences, we note that despite these differences in annotation

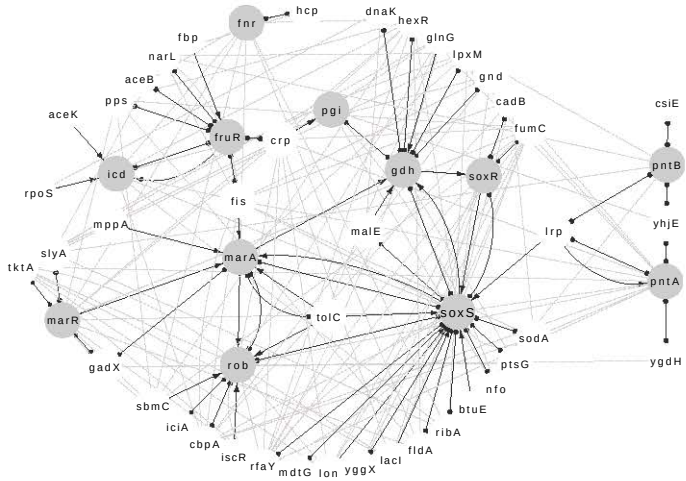


Figure 7.3: The complete network obtained from EVEX (solid lines) and microarray-based co-expression analysis (dashed lines). In the EVEX network, circle-terminated connections indicate binding and arrows indicate regulation. The key genes are highlighted in gray; 2 key genes are not present since no relevant events were available in EVEX. Note that only events involving at least one key gene are extracted, therefore no events between candidate genes are currently shown.

schemes, all of the highest confidence events are semantically correct and equivalent to the KEGG interaction type.

We have thus illustrated that EVEX allows for fast extraction of relevant literature and interaction partners of uniquely identified genes. Clearly, this type of text mining retrieval can assist pathway curators by helping their search for relevant candidate interactions and articles. In future work, we will further focus on these opportunities, integrating data from various resources with text mining information and analysing the resulting molecular networks in a systems biology study.

7.3.5 Network analysis

Finally, the EVEX dataset as well as the associated web application have been applied in a focused study targeting the regulation of NADP(H) expression in *E. coli*, with encouraging results. The Ensembl Genomes generalization was used to allow for homology-based inference and the regulatory network extracted from EVEX was integrated with microarray co-expression data.

The starting point for this study was a list of 14 key genes known to be involved in NADP(H)-metabolism. From EVEX, 348 unique events were extracted involving at least one of these key genes or one of its orthologues, as defined by Ensembl Genomes.

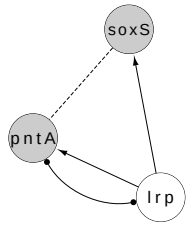


Figure 7.4: Triangular pattern found in the integrated networks of EVEX (solid lines) and microarray-based co-expression analysis (dashed lines).

In total, these events involve 152 unique families. A manual curation step was necessary to distinguish the true events from the wrong ones, resulting in a clean set of 169 correct events. Additionally, 36 events were extracted with an incorrect type, either through label substitution of regulation vs. binding, or constituting a relationship which does not have an appropriate type defined in the event representation. Three more events were found encoding a regulation in the opposite direction (i.e. Cause and Theme argument switched). The final set of events then consists of true positives and corrected false positives, and is further cleaned by removing downstream regulatory events of the key genes. From the resulting set of 118 events, 81 events were judged to be correctly assigned to gene families, resulting in a list of 41 unique regulators of one of the 14 key genes (and their homologs).

The manual evaluation of the initial set of events to construct the EVEX network amounted to a little less than three days of work of one person. Of the two validation steps (event correctness and family assignment), evaluating the correctness of the family assignments was clearly the more labor intensive one, as it often required careful identification of the species, strain, and sub-strain involved. However, we consider the manual evaluation step of great importance and not excessively labor-intensive, particularly compared to the effort that would be necessary to build such a network without any text mining support.

The final, clean EVEX network was integrated with specific microarray data, selected based on their expected relevance for NADP(H)-metabolism in *E. coli* (Figure 7.3). In this integrated network, several interesting patterns can be found, indicating that apparent indirect interactions are in fact direct, or vice versa. For example, the co-expression between *pntA* and *soxS* is explained by co-regulation of the *lrp* regulon, as extracted from text (Figure 7.4).

Finally, it is possible to only investigate candidate regulators found in non-*E. coli* literature, in an effort to capture unknown information that could be retrieved through the homology-based event generalizations (Section 6.4.4). Of the 41 candidate regulators found previously, 5 originate only from non-*E. coli* studies. These genes and their interconnectivity in the network was extensively studied, and *hexR* selected as a promis-

ing candidate for further research. These findings illustrated how EVEX can be used in real-world biological use cases.

7.4 Conclusion

This thesis presents several novel text mining extraction techniques, with applications ranging from the extraction of protein-protein interactions to regulatory events and non-causal relations. Throughout our studies, thorough evaluations have been a key aspect. We have presented several detailed analyses on parameter optimization and have critically assessed the different modules in the presented frameworks. Further, we have conducted large-scale cross-domain evaluations. We believe these detailed evaluations and their conclusions to be valuable for the entire BioNLP community.

Further, a bibliome-wide resource has been introduced containing text mining information from all PubMed abstracts and PMC Open Access full-text articles. This rich resource contains more than 34 million events and has been fully integrated with gene-family definitions from HomoloGene and Ensembl, and full gene normalization to unique Entrez Gene IDs. Both a publicly available MySQL database as well as an intuitive web interface were developed to support this EVEX resource. Finally, EVEX has been shown useful for pathway and database curation as well as network analysis, hypothesis generation and knowledge discovery. During these assessments we have illustrated the added value of processing full-text, underlining the importance of extending the Open Access set of PubMed Central.

In future work, we will build upon the preliminary results presented here, combining the EVEX text mining predictions with other biological data in a series of focused use cases. To enable application in practical use cases, we will extend the web application and provide support for manual evaluation of the EVEX data and for exporting specific events to tab delimited formats or cytoscape networks. To further extend the semantic scope of the EVEX dataset, we will integrate new extraction targets, such as large-scale detection of entity relations and inclusion of epigenetics data.

A
Lexicon and acronyms

This appendix summarizes the acronyms used throughout this thesis and additionally provides a lexicon on domain-specific terminology, explaining the usage of the word within the context of this work.

A

API	Application programming interface
Annotation	Meta-data provided by human annotators, typically marking linguistic and semantic information in a corpus
Argument	A participant of a certain biomolecular event, either theme or cause

B

BC	BioCreative
BioCreative	A community-wide challenge on PPI extraction and gene normalization
BioNLP	The research domain of natural language processing for biomedical texts
BOW	Bag-of-words approach, listing relevant words as an unordered collection

C

Cause	The argument of an event which drives the action (affector)
CV	Cross validation
COALS	*Correlated accurrence analogue to lexical semantics*, a word space model
Corpus	A collection of annotated documents
Cross validation	An evaluation technique that divides the training data in separate training/test subsets for fine-tuning data mining models

D

DB	Database
DNA	Deoxyribonucleic acid

E

EG	Entrez Gene
ENS	Ensembl
Ensembl	A resource of genome databases for eukaryots
Ensembl genomes	Extension of Ensembl to metazoa, plants, protists, fungi, and bacteria
Entity relation	A non-causal relation between two entities
Entrez Gene	A gene-centered resource hosted by NCBI
EPI	Epigenetics and post-translational modifications task of the BioNLP Shared Task of 2011
EVEX	A bibliome-wide text mining resource for event extraction

F

False negative	A missing prediction
False positive	A wrong prediction
Feature selection	A technique for reducing model complexity
FN	False negative
FP	False positive
FS	Feature selection
FTP	File transfer protocol

G

Gene Ontology	A structured vocabulary for annotating gene functions, biological processes and cellular components
GGP	Gene or gene product
GNR	Gene name recognition
GO	Gene Ontology

H

HAL	*Hyperspace analogue to language*, a word space model
HomoloGene	A resource of eukaryotic gene families
HG	HomoloGene
HPRD	*Human Protein Reference Database*, a protein-centered resource
HTML	Hyptertext markup language

I

ID	Identifier
IE	Information extraction
IEPA	*Interaction extraction performance assessment*, a PPI corpus
Information extraction	The text mining subtask of extracting structured information (events, relations) from unstructured text
Information retrieval	The text mining subtask of retrieving relevant documents for information extraction
IR	Information retrieval

L

LLL	*Learning language in logic*, a PPI corpus
LSA	*Latent semantic analysis*, a word space model

M

Markov clustering	An unsupervised clustering algorithm
MCL	Markov clustering

MGI	*Mouse Genome Informatics*, the authoritative resource for genetic information on mouse
ML	Machine learning

N

Named entity	A mention of a specific object in text (e.g. a person, gene, organism, ...)
Named entity recognition	The text mining subtask of identifying named entities in text
Named entity normalization	The text mining subtask of assigning a unique identifier to a named entity
Natural language processing	The research domain of automated processing of human languages
NCBI	*National Center for Biotechnology Information*, an important access portal for various genetic data resources
NE	Named entity
NER	Named entity recognition
NEN	Named entity normalization
NI	Negative instance, a false classification example
NLP	Natural language processing
Normalization	cf. named entity normalization

0

OA	Open access
OBO	*Open Biomedical Ontologies*, a resource of structured vocabularies in the biomedical field
Open Access	Unrestricted access and reuse

P

PDF	Portable document format
PLEV	*Plant evaluation*, a text mining evaluation dataset
PM	PubMed
PMC	PubMed Central
P3DB	Plant protein phosphorylation database
PP	Percentage point
PPI	Protein-protein interaction

PTM	Post-translational modification
PubMed	A resource of biomedical citations
PubMed Central	A resource of full-text biomedical articles

R

Radial basis function	A type of SVM kernel
RBF	Radial basis function
RI	*Random indexing*, a word space model
RNA	Ribonucleic acid

S

ST	Shared task
SVM	Support vector machine
Support vector machine	A classification method that constructs a hyperplane to separate data points with different class labels in a multidimensional space

R

Radial basis function	An SVM kernel, usually based on a Gaussian function
RBF	Radial basis function
REL	Entity relation

T

TAIR	*The Arabidopsis Information Resource*, the authoritative resource for genetic information on *Arabidopsis thaliana*
TEES	Turku Event Extraction System
Theme	The argument of an event which undergoes the action (affectee)
TN	True negative
TP	True positive
True negative	An instance correctly predicted as being false
True positive	A correct prediction

U

UniProt	A protein-centered resource

X

XML	Extensible markup language

W

Word space model	A method for deriving word meaning from lexical co-occurrence

B
Publications

Journal publications

- **Van Landeghem, S.**, Hakala, K., Rönnqvist, S., Salakoski, T., Van de Peer, Y. and Ginter, F. (2012). Exploring biomolecular literature with EVEX: Connecting genes through events, homology and indirect associations. *Advances in Bioinformatics*, in press.
- **Van Landeghem, S.**, Björne, J., Abeel, T., De Baets, B., Salakoski, T., Van de Peer, Y. (2012). Semantically linking molecular entities in literature through entity relationships. *BMC Bioinformatics* 13 (Suppl. 8): S6.
- **Van Landeghem, S.**, De Baets, B., Van de Peer, Y., Saeys, Y. (2011). High-precision bio-molecular event extraction from text using parallel binary classifiers. *Computational Intelligence* 27(24), p. 645-664.
- Kano, Y., Björne, J., Ginter, F., Salakoski, T., Buyko, E., Hahn, U., Cohen, K., Verspoor, K., Roeder, C., Hunter, L., Kilicoglu, H., Bergler, S., **Van Landeghem, S.**, Van Parys, T., Van de Peer, Y., Miwa, M., Ananiadou, S., Neves, M., Pascual-Montano, A., Özgür, A., Radev, D., Riedel, S., Sætre, R., Chun, H.W., Kim, J.D., Pyysalo, S., Ohta, T., Tsujii, J. (2011). U-Compare bio-event meta-service: compatible BioNLP event extraction services. *BMC Bioinformatics* 12:481.
- **Van Landeghem, S.***, Abeel, T.*, Saeys, Y., Van de Peer, Y. (2010). Discriminative and informative features for biomolecular text mining with ensemble feature selection. *Bioinformatics* 26 (18): i554-i560.
  *: contributed equally
- Saeys, Y., **Van Landeghem, S.**, Van de Peer, Y. (2010). Event based text mining for

integrated network construction. *Journal of Machine Learning Research, workshop and conference proceedings* 8, p. 112-121.

Conference publications

- Kaewphan, S., Peltonen, S., **Van Landeghem, S.**, Van de Peer, Y., Jones, P., Ginter, F. (2012). Integrating large-scale text mining and co-expression networks: Targeting NADP(H) metabolism in E. coli with event extraction. *LREC workshop on Building and Evaluating Resources for Biomedical Text Mining*, in press. Istanbul, Turkey.
- **Van Landeghem, S.**, Ginter, F., Van de Peer, Y., Salakoski, T. (2011). EVEX: A PubMed-scale resource for homology-based generalization of text mining predictions. *BioNLP Workshop*, p. 28-37. Portland, Oregon, USA.
- **Van Landeghem, S.**, Pyysalo, S., Ohta, T., Van de Peer, Y. (2010). Integration of static relations to enhance event extraction from text. Proceedings of *BioNLP Workshop*, p. 144-152. Uppsala, Sweden.
- Saeys, Y., **Van Landeghem, S.**, Van de Peer, Y. (2009). Integrated network construction using event based text mining. *Machine Learning in Systems Biology workshop (MLSB)*, p. 105-114. Ljubljana, Slovenia.
- **Van Landeghem, S.**, Saeys, Y., De Baets, B., Van de Peer, Y. (2009). Analyzing text in search of biomolecular events: a high-precision machine learning framework. *BioNLP Shared Task Workshop*, p. 128-136. Colorado, USA.
- **Van Landeghem, S.**, Saeys, Y., De Baets, B., Van de Peer, Y. (2008). Extracting protein-protein interactions from text using rich feature vectors and feature selection. *International Symposium on Semantic Mining in Biomedicine (SMBM)*, p. 77-84. Turku, Finland.

Editorial

- Abeel, T.*, **Van Landeghem, S.***, Morante, R., Van Asch, V., Van de Peer, Y., Daelemans, W., Saeys, Y. (2010). Highlights of the BioTM 2010 workshop on advances in bio text mining. *BMC Bioinformatics*, 11, I1.
  *: contributed equally

Abstracts of oral presentations

- **Van Landeghem, S.**, Saeys, Y., De Baets, B., Van de Peer, Y. (2008). Applying supervised learning and feature selection techniques to extract genetic interactions from text. *Benelux Bioinformatics Conference (BBC)*, p. 38-39. Maastricht, The Netherlands.
- **Van Landeghem, S.**, Saeys, Y., De Baets, B., Van de Peer, Y. (2008). Benchmarking machine learning techniques for the extraction of protein-protein interactions from text. *Benelearn*, p. 79-80. Spa, Belgium.

Abstracts of poster presentations

- **Van Landeghem, S.**, Abeel, T., De Baets, B., Van de Peer, Y. (2011). Detecting entity relations as a supporting task for bio-molecular event extraction. *BioNLP Shared Task Workshop*, p. 147-148. Oregon, USA.
- **Van Landeghem, S.**, Pyysalo, S., Ohta, T., Van de Peer, Y. (2010). Towards a more detailed representation of biomolecular text mining results. *European Student Council Symposium (ESCS)*. Ghent, Belgium.
- **Van Landeghem, S.**, De Baets, B., Van de Peer, Y., Saeys, Y. (2009). Summarizing bio-molecular interactions from research articles using advanced text mining. *Benelux Bioinformatics Conference (BBC)*. Liege, Belgium.
- **Van Landeghem, S.**, De Baets, B., Van de Peer, Y., Saeys, Y. (2009). Training a text miner to summarize bio-molecular interactions found in research articles. *Machine Learning Summerschool*. Cambridge, UK.
- **Van Landeghem, S.**, Saeys, Y., De Baets, B., Van de Peer, Y. (2008). Applying supervised learning and feature selection techniques to extract genetic interactions from text. *CIL PhD students day @ ECML*. Antwerp, Belgium.
- **Van Landeghem, S.**, Saeys, Y., Cornelis, C., Benotmane, M.A., Van de Peer, Y. (2007). GeneFetch: enriching gene information with text mining for automatic knowledge base construction. *International Workshop on Machine Learning in Systems Biology (MLSB)*, p105. Evry, France.

References

Abeel, T., Helleputte, T., Van de Peer, Y., Dupont, P., and Saeys, Y. (2010). Robust biomarker identification for cancer diagnosis with ensemble feature selection methods. *Bioinformatics*, **26**(3), 392–398.

Agarwal, S., Yu, H., and Kohane, I. (2011). BioNOT: A searchable database of biomedical negated sentences. *BMC Bioinformatics*, **12**(1), 420.

Arighi, C., Lu, Z., Krallinger, M., Cohen, K., Wilbur, J., Valencia, A., Hirschman, L., and Wu, C. (2011). Overview of the BioCreative III workshop. *BMC Bioinformatics*, **12**(Suppl 8), S1.

Baginsky, S., Hennig, L., Zimmermann, P., and Gruissem, W. (2010). Gene expression analysis, proteomics, and network discovery. *Plant Physiology*, **152**(2), 402–410.

Baumgartner, W., Lu, Z., Johnson, H., Caporaso, J. G., Paquette, J., Lindemann, A., White, E., Medvedeva, O., Cohen, K. B., and Hunter, L. (2008). Concept recognition for extracting protein interaction relations from biomedical text. *Genome Biology*, **9**(Suppl 2), S9.

Bernstein, B. E., Meissner, A., and Lander, E. S. (2007). The mammalian epigenome. *Cell*, **128**(4), 669 – 681.

Bikel, D. M. (2004). Intricacies of collins' parsing model. *COMPUTATIONAL LINGUISTICS*, **30**, 479–511.

Björne, J. and Salakoski, T. (2011). Generalizing biomedical event extraction. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 183–191.

Björne, J., Heimonen, J., Ginter, F., Airola, A., Pahikkala, T., and Salakoski, T. (2009). Extracting complex biological events with rich graph-based feature sets. In *Proceedings of the BioNLP 2009 Workshop*, pages 10–18.

Björne, J., Ginter, F., Pyysalo, S., Tsujii, J., and Salakoski, T. (2010). Scaling up biomedical event extraction to the entire PubMed. In *Proceedings of the BioNLP 2010 Workshop*, pages 28–36.

Björne, J., Ginter, F., and Salakoski, T. (2012). Generalizing biomedical event extraction. *BMC Bioinformatics*, **13**(suppl. 8), S4.

Blake, J. A., Bult, C. J., Kadin, J. A., Richardson, J. E., Eppig, J. T., and the Mouse Genome Database Group (2011). The Mouse Genome Database (MGD): premier model organism resource for mammalian genomics and genetics. *Nucleic Acids Research*, **39**(suppl 1), D842–D848.

Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, pages 144–152.

Buckland, M. and Gey, F. (1994). The relationship between recall and precision. *Journal of the American Society for Information Science*, **45**(1), 12–19.

Bunescu, R., Ge, R., Kate, R. J., Marcotte, E. M., Mooney, R. J., Ramani, A. K., and Wong, Y. W. (2005). Comparative experiments on learning information extractors for proteins and their interactions. *Artificial Intelligence in Medicine*, **33**(2), 139–155.

Burgess, C. and Lund, K. (1997). Modelling parsing constraints with high-dimensional context space. *Language and Cognitive Processes*, **12**, 177–210.

Buyko, E., Faessler, E., Wermter, J., and Hahn, U. (2009). Event extraction from trimmed dependency graphs. In *Proceedings of the BioNLP 2009 Workshop*, pages 19–27.

Caporaso, J. G., Deshpande, N., Fink, J. L., Bourne, P. E., Cohen, K. B., and Hunter, L. (2008). Intrinsic evaluation of text mining tools may not predict performance on realistic tasks. *Pacific Symposium on Biocomputing*, pages 640–651.

Carballo, J. and Cha, R. (2007). Meiotic roles of Mec1, a budding yeast homolog of mammalian ATR/ATM. *Chromosome Research*, **15**(5), 539–550.

Chang, C.-C. and Lin, C.-J. (2001). *LIBSVM: a library for support vector machines*. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Chen, L., Liu, H., and Friedman, C. (2005). Gene name ambiguity of eukaryotic nomenclatures. *Bioinformatics*, **21**, 248–256.

Cohen, K. B., Verspoor, K., Johnson, H. L., Roeder, C., Ogren, P. V., Baumgartner, Jr., W. A., White, E., Tipney, H., and Hunter, L. (2009). High-precision biological event extraction with a concept recognizer. In *Proceedings of the BioNLP 2009 Workshop*, pages 50–58.

Cohen, K. B., Johnson, H., Verspoor, K., Roeder, C., and Hunter, L. (2010). The structural and content aspects of abstracts versus bodies of full text journal articles are different. *BMC Bioinformatics*, **11**(1), 492.

De Bodt, S., Carvajal, D., Hollunder, J., Van den Cruyce, J., Movahedi, S., and Inzé, D. (2010). CORNET: A user-friendly tool for data mining and integration. *Plant Physiology*, **152**(3), 1167–1179.

de Marneffe, M., Maccartney, B., and Manning, C. (2006). Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC-06*, pages 449–454.

de Marneffe, M.-C. and Manning, C. D. (2011). *Stanford typed dependencies manual*. Stanford University, Technical report.

Ding, J., Berleant, D., Nettleton, D., and Wurtele, E. (2002). Mining Medline: Abstracts, sentences, or phrases? In *Pacific Symposium on Biocomputing*, pages 326–337.

Erkan, G., Ozgur, A., and Radev, D. (2007). Extracting interacting protein pairs and evidence sentences by using dependency parsing and machine learning techniques. In *Proceedings of BioCreative 2*, pages 326–337.

Fayruzov, T., De Cock, M., Cornelis, C., and Hoste, V. (2008). DEEPER: a full parsing based approach to protein relation extraction. In *Proceedings of the 6th European conference on Evolutionary computation, machine learning and data mining in bioinformatics*, pages 36–47.

Flicek, P., Amode, M. R., Barrell, D., Beal, K., Brent, S., Chen, Y., Clapham, P., Coates, G., Fairley, S., Fitzgerald, S., Gordon, L., Hendrix, M., Hourlier, T., Johnson, N., Kähäri, A., Keefe, D., Keenan, S., Kinsella, R., Kokocinski, F., Kulesha, E., Larsson, P., Longden, I., McLaren, W., Overduin, B., Pritchard, B., Riat, H. S. S., Rios, D., Ritchie, G. R., Ruffier, M., Schuster, M., Sobral, D., Spudich, G., Tang, Y. A., Trevanion, S., Vandrovcova, J., Vilella, A. J., White, S., Wilder, S. P., Zadissa, A., Zamora, J., Aken, B. L., Birney, E., Cunningham, F., Dunham, I., Durbin, R., Fernández-Suarez, X. M., Herrero, J., Hubbard, T. J., Parker, A., Proctor, G., Vogel, J., and Searle, S. M. (2011). Ensembl 2011. *Nucleic Acids Research*, **39**(Database issue).

Fulton, T. M., Van der Hoeven, R., Eannetta, N. T., and Tanksley, S. D. (2002). Identification, analysis, and utilization of conserved ortholog set markers for comparative genomics in higher plants. *Plant Cell*, **14**(5), 1457–1467.

Fundel, K., Küffner, R., and Zimmer, R. (2006). RelEx–relation extraction using dependency parse trees. *Bioinformatics*, **23**(3), 365–371.

Gao, J., Agrawal, G. K., Thelen, J. J., and Xu, D. (2008). P3DB: a plant protein phosphorylation database. *Nucleic Acids Research*, **37**, D960–962.

Gerner, M., Nenadic, G., and Bergman, C. M. (2010). LINNAEUS: a species name identification system for biomedical literature. *BMC Bioinformatics*, **11**(1).

Ghosh, S., Matsuoka, Y., Asai, Y., Hsin, K.-Y. Y., and Kitano, H. (2011). Software for systems biology: from tools to integrated platforms. *Nature reviews. Genetics*, **12**(12), 821–832.

Giuliano, C., Lavelli, A., and Romano, L. (2006). Exploiting shallow linguistic information for relation extraction from biomedical literature. In *Proceedings of EACL 2006*, pages 401–408.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, **3**, 1157–1182.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA data mining software: an update. *SIGKDD Explorations Newsletter*, **11**, 10–18.

Hearst, M. A., Divoli, A., Guturu, H., Ksikes, A., Nakov, P., Wooldridge, M. A., and Ye, J. (2007). BioText Search Engine: beyond abstract search. *Bioinformatics*, **23**, 2196–2197.

Hirschman, L., Yeh, A., Blaschke, C., and Valencia, A. (2005). Overview of BioCreAtIvE: critical assessment of information extraction for biology. *BMC Bioinformatics*, **6**(Suppl 1), S1.

Hoffmann, R. and Valencia, A. (2004). A gene network for navigating the literature. *Nature genetics*, **36**(7).

Hulsen, T., Huynen, M., de Vlieg, J., and Groenen, P. (2006). Benchmarking ortholog identification methods using functional genomics data. *Genome Biology*, **7**(4).

Jensen, L. J., Kuhn, M., Stark, M., Chaffron, S., Creevey, C., Muller, J., Doerks, T., Julien, P., Roth, A., Simonovic, M., Bork, P., and von Mering, C. (2009). STRING 8–a global view on proteins and their functional interactions in 630 organisms. *Nucleic Acids Research*, **37**(suppl 1), D412–D416.

Jiang, J. and Zhai, C. (2007). A Systematic Exploration of the Feature Space for Relation Extraction. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 113–120.

Jurgens, D. and Stevens, K. (2010). The S-Space package: an open source package for word space models. In *Proceedings of the ACL 2010 System Demonstrations*, pages 30–35.

Kaewphan, S., Kreula, S., Van Landeghem, S., Van de Peer, Y., Jones, P., and Ginter, F. (2012). Integrating large-scale text mining and co-expression networks: Targeting NADP(H) metabolism in E. coli with event extraction. In *Proceedings of the Third Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM 2012)*.

Kanehisa, M. and Goto, S. (2000). KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, **28**, 27–30.

Katrenko, S. and Adriaans, P. (2007). Learning relations from biomedical corpora using dependency trees. In *Proceedings of the 1st international conference on Knowledge discovery and emergent complexity in bioinformatics*, pages 61–80.

Kazama, J. and Tsujii, J. (2003). Evaluation and extension of maximum entropy models with inequality constraints. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 137–144.

Kersey, P. J., Lawson, D., Birney, E., Derwent, P. S., Haimel, M., Herrero, J., Keenan, S., Kerhornou, A., Koscielny, G., Khri, A., Kinsella, R. J., Kulesha, E., Maheswari, U., Megy, K., Nuhn, M., Proctor, G., Staines, D., Valentin, F., Vilella, A. J., and Yates, A. (2010). Ensembl Genomes: Extending ensembl across the taxonomic space. *Nucleic Acids Research*, **38**(suppl 1), D563–D569.

Kilicoglu, H. and Bergler, S. (2011). Adapting a general semantic interpretation approach to biological event extraction. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 173–182.

Kim, J. D., Ohta, T., and Tsujii, J. (2008a). Corpus annotation for mining biomedical events from literature. *BMC Bioinformatics*, **9**(1).

Kim, J.-D., Ohta, T., Pyysalo, S., Kano, Y., and Tsujii, J. (2009). Overview of BioNLP'09 shared task on event extraction. In *Proceedings of the BioNLP 2009 Workshop*, pages 1–9.

Kim, J.-D., Pyysalo, S., Ohta, T., Bossy, R., Nguyen, N., and Tsujii, J. (2011). Overview of BioNLP Shared Task 2011. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 1–6.

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

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

Krallinger, M., Morgan, A., Smith, L., Leitner, F., Tanabe, L., Wilbur, J., Hirschman, L., and Valencia, A. (2008). Evaluation of text-mining systems for biology: overview of the second BioCreative community challenge. *Genome Biology*, **9**(Suppl 2), S1.

Kuncheva, L. (2007). A stability index for feature selection. In *Proceedings of the 25th International Multi-Conference on Artificial Intelligence and Applications*, pages 390–395.

Lamesch, P., Berardini, T. Z., Li, D., Swarbreck, D., Wilks, C., Sasidharan, R., Muller, R., Dreher, K., Alexander, D. L., Garcia-Hernandez, M., Karthikeyan, A. S., Lee, C. H., Nelson, W. D., Ploetz, L., Singh, S., Wensel, A., and Huala, E. (2012). The Arabidopsis Information Resource (TAIR): improved gene annotation and new tools. *Nucleic Acids Research*, **40**(D1), D1202–D1210.

Landauer, T. K. and Dumais, S. T. (1997). A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, **104**, 211–240.

Larrañaga, P., Calvo, B., Santana, R., Bielza, C., Galdiano, J., Inza, I., Lozano, J. A., Armañanzas, R., Santafé, G., Pérez, A., and Robles, V. (2006). Machine learning in bioinformatics. *Briefings in Bioinformatics*, **7**(1), 86–112.

Leaman, R. and Gonzalez, G. (2008). BANNER: an executable survey of advances in biomedical named entity recognition. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, pages 652–663.

Leitner, F., Mardis, S., Krallinger, M., Cesareni, G., Hirschman, L., and Valencia, A. (2010). An overview of BioCreative II.5. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, **7**(3), 385–399.

Lewis, R. M. and Torczon, V. (1999). Pattern search algorithms for bound constrained minimization. *SIAM Journal on Optimization*, **9**(4), 1082–1099.

Li, L., Stoeckert, C. J., and Roos, D. S. (2003). OrthoMCL: identification of ortholog groups for eukaryotic genomes. *Genome research*, **13**(9), 2178–2189.

Loewenstein, Y., Raimondo, D., Redfern, O. C., Watson, J., Frishman, D., Linial, M., Orengo, C., Thornton, J., and Tramontano, A. (2009). Protein function annotation by homology-based inference. *Genome biology*, **10**(2), 207.

Lu, Z., Kao, H. Y., Wei, C. H., Huang, M., Liu, J., Kuo, C. J., Hsu, C. N., Tsai, R., Dai, H. J., Okazaki, N., Cho, H. C., Gerner, M., Solt, I., Agarwal, S., Liu, F., Vishnyakova, D., Ruch, P., Romacker, M., Rinaldi, F., Bhattacharya, S., Srinivasan, P., Liu, H., Torii, M., Matos, S., Campos, D., Verspoor, K., Livingston, K., and Wilbur, W. (2011). The gene normalization task in BioCreative III. *BMC Bioinformatics*, **12**(Suppl 8), S2.

McClosky, D., Charniak, E., and Johnson, M. (2006). Reranking and self-training for parser adaptation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 337–344.

Monard, M. C. and Batista, G. (2003). Learning with skewed class distributions. *Learning*, pages 1–9.

Nedellec, C. (2006). Learning language in logic - genic interaction extraction challenge. In *Proceedings of LLL'05*, pages 31–37.

Ohta, T., Miyao, Y., Ninomiya, T., Tsuruoka, Y., Yakushiji, A., Masuda, K., Takeuchi, J., Yoshida, K., Hara, T., Kim, J.-D., Tateisi, Y., and Tsujii, J. (2006). An intelligent search engine and GUI-based efficient MEDLINE search tool based on deep syntactic parsing. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 17–20.

Ohta, T., Pyysalo, S., Jin-Dong, K., and Tsujii, J. (2009). A re-evaluation of biomedical named entity - term relations. In *Proceedings of LBM'09*.

Ohta, T., Pyysalo, S., and Tsujii, J. (2011). From pathways to biomolecular events: Opportunities and challenges. In *Proceedings of the BioNLP 2011 Workshop*, pages 105–113.

Okazaki, N. and Tsujii, J. (2010). Simple and efficient algorithm for approximate dictionary matching. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 851–859.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, **14**(3), 130–137.

Proost, S., Van Bel, M., Sterck, L., Billiau, K., Van Parys, T., Van de Peer, Y., and Vandepoele, K. (2009). PLAZA: A comparative genomics resource to study gene and genome evolution in plants. *Plant Cell*, **21**(12), 3718–3731.

Pyysalo, S., Ginter, F., Heimonen, J., Björne, J., Boberg, J., Järvinen, J., and Salakoski, T. (2007). BioInfer: a corpus for information extraction in the biomedical domain. *BMC Bioinformatics*, **8**(1), 50+.

Pyysalo, S., Airola, A., Heimonen, J., Björne, J., Ginter, F., and Salakoski, T. (2008). Comparative analysis of five protein-protein interaction corpora. *BMC Bioinformatics*, **9**(S6).

Pyysalo, S., Ohta, T., Kim, J.-D., and Tsujii, J. (2009). Static relations: a piece in the biomedical information extraction puzzle. In *Proceedings of the BioNLP 2009 Workshop*, pages 1–9.

Pyysalo, S., Ohta, T., and Tsujii, J. (2011). Overview of the entity relations (REL) supporting task of BioNLP Shared Task 2011. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 83–88.

Rebholz-Schuhmann, D., Kirsch, H., Arregui, M., Gaudan, S., Riethoven, M., and Stoehr, P. (2007). EBIMed - text crunching to gather facts for proteins from Medline. *Bioinformatics*, **23**(2), e237–e244.

Riedel, S. and McCallum, A. (2011). Robust biomedical event extraction with dual decomposition and minimal domain adaptation. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 46–50.

Rohde, D. L. T., Gonnerman, L. M., and Plaut, D. C. (2006). An improved model of semantic similarity based on lexical co-occurence. *Communications of the ACM*, **8**, 627–633.

Saetre, R., Sagae, K., and Tsujii, J. (2008). Syntactic features for protein-protein interaction extraction. In *Proceedings of the Second International Symposium on Languages in Biology and Medicine (LBM2007)*.

Saeys, Y., Inza, I., and Larranaga, P. (2007). A review of feature selection techniques in bioinformatics. *Bioinformatics*, **23**(19), 2507–2517.

Saeys, Y., Abeel, T., and Van de Peer, Y. (2008). Robust feature selection using ensemble feature selection techniques. In *Proceedings of ECML/PKDD*, volume 5212, pages 313–325.

Sahlgren, M., Holst, A., and Kanerva, P. (2008). Permutations as a means to encode order in word space. In *Proceedings of the 30th Annual Conference of the Cognitive Science Society*, pages 1300–1305. Cognitive Science Society.

Santorini, B. (1990). Part-of-speech tagging guidelines for the penn treebank project (3rd revision). *Technical Reports CIS*.

Sayers, E. W., Barrett, T., Benson, D. A., Bolton, E., Bryant, S. H., Canese, K., Chetvernin, V., Church, D. M., DiCuccio, M., Federhen, S., Feolo, M., Geer, L. Y., Helmberg, W., Kapustin, Y., Landsman, D., Lipman, D. J., Lu, Z., Madden, T. L., Madej, T., Maglott, D. R., Marchler-Bauer, A., Miller, V., Mizrachi, I., Ostell, J., Panchenko, A., Pruitt, K. D., Schuler, G. D., Sequeira, E., Sherry, S. T., Shumway, M., Sirotkin, K., Slotta, D., Souvorov, A., Starchenko, G., Tatusova, T. A., Wagner, L., Wang, Y., Wilbur, J., Yaschenko, E., and Ye, J. (2010). Database resources of the national center for biotechnology information. *Nucleic Acids Research*, **38**(suppl 1), D5–D16.

Segal, E., Shapira, M., Regev, A., Pe'er, D., Botstein, D., Koller, D., and Friedman, N. (2003). Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature Genetics*, **34**(2), 166–176.

Stark, C., Breitkreutz, B.-J., Chatr-aryamontri, A., Boucher, L., Oughtred, R., Livstone, M. S., Nixon, J., Van Auken, K., Wang, X., Shi, X., Reguly, T., Rust, J. M., Winter, A., Dolinski, K., and Tyers, M. (2011). The BioGRID interaction database: 2011 update. *Nucleic Acids Research*, **39**(suppl 1), D698–D704.

Stenetorp, P., Topić, G., Pyysalo, S., Ohta, T., Kim, J.-D., and Tsujii, J. (2011a). BioNLP Shared Task 2011: Supporting resources. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 112–120.

Stenetorp, P., Topić, G., Pyysalo, S., Ohta, T., Kim, J.-D., and Tsujii, J. (2011b). BioNLP Shared Task 2011: Supporting resources. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 112–120.

Tanabe, L., Xie, N., Thom, L. H., Matten, W., and Wilbur, J. (2005). GENETAG: a tagged corpus for gene/protein named entity recognition. *BMC Bioinformatics*, **6 Suppl 1**.

The Gene Ontology Consortium (2008). The Gene Ontology project in 2008. *Nucleic Acids Research*, **36**(Database issue), D440–D444.

The UniProt Consortium (2011). Ongoing and future developments at the universal protein resource. *Nucleic Acids Research*, **39**(suppl 1), D214–D219.

van Dongen, S. (2000). *Graph clustering by flow simulation*. Ph.D. thesis, University of Utrecht.

Van Landeghem, S., Saeys, Y., De Baets, B., and Van de Peer, Y. (2008). Extracting protein-protein interactions from text using rich feature vectors and feature selection. In *Proceedings of the Third International Symposium on Semantic Mining in Biomedicine (SMBM)*, pages 77–84.

Van Landeghem, S., Saeys, Y., De Baets, B., and Van de Peer, Y. (2009). Analyzing text in search of bio-molecular events: a high-precision machine learning framework. In *Proceedings of the BioNLP 2009 Workshop*, pages 128–136.

Van Landeghem, S., Abeel, T., Saeys, Y., and Van de Peer, Y. (2010). Discriminative and informative features for biomolecular text mining with ensemble feature selection. *Bioinformatics*, **26**(18), i554–i560.

Van Landeghem, S., Abeel, T., De Baets, B., and Van de Peer, Y. (2011a). Detecting entity relations as a supporting task for bio-molecular event extraction. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pages 147–148.

Van Landeghem, S., Ginter, F., Van de Peer, Y., and Salakoski, T. (2011b). EVEX: a PubMed-scale resource for homology-based generalization of text mining predictions. In *Proceedings of the BioNLP 2011 Workshop*, pages 28–37.

Van Landeghem, S., De Baets, B., Van de Peer, Y., and Saeys, Y. (2011c). High-precision bio-molecular event extraction from text using parallel binary classifiers. *Computational Intelligence*, **27**(4), 546–664.

Van Landeghem, S., Hakala, K., Rönnqvist, S., Salakoski, T., Van de Peer, Y., and Ginter, F. (2012a). Exploring biomolecular literature with EVEX: Connecting genes through events, homology and indirect associations. *Advances in Bioinformatics*. In press.

Van Landeghem, S., Björne, J., Abeel, T., De Baets, B., Salakoski, T., and Van de Peer, Y. (2012b). Semantically linking molecular entities in literature through entity relationships. *BMC Bioinformatics*, **13**(Suppl 8), S6.

Wang, H., Huang, M., Ding, S., and Zhu, X. (2008). Exploiting and integrating rich features for biological literature classification. *BMC Bioinformatics*, **9**(Suppl 3).

Wei, C.-H. and Kao, H.-Y. (2011). Cross-species gene normalization by species inference. *BMC Bioinformatics*, **12**(Suppl 8), S5.

Xu, S., McCusker, J., and Krauthammer, M. (2008). Yale Image Finder (YIF): a new search engine for retrieving biomedical images. *Bioinformatics*, **24**, 1968–1970.

Zhu, X., Gerstein, M., and Snyder, M. (2007). Getting connected: analysis and principles of biological networks. *Genes & Development*, **21**(9), 1010–1024.