Tapio Salakoski
Dietrich Rebholz-Schuhmann
Sampo Pyysalo (Eds.)

Proceedings of

SMBM 2008

The Third International Symposium on Semantic Mining in Biomedicine

Turku Centre *for* Computer Science

TUCS General Publication
No 51, September 2008

Proceedings of

SMBM 2008

The Third International Symposium on Semantic Mining in Biomedicine

September 1st - 3rd, 2008, Turku, Finland

Editors:

Tapio Salakoski
Dietrich Rebholz-Schuhmann
Sampo Pyysalo

TUCS General Publication
No 51, September 2008
Digital Edition

Foreword to the third International Symposium for Semantic Mining in Biomedicine (SMBM 2008)

Today, the importance of semantic mining in support of biomedical domain research is increasingly recognized. The scientific community strives to improve access to the large and rapidly growing domain literature and support database curation efforts. An emerging trend is to build toward text mining systems capable of assisting in hypothesis generation that could be integrated in genome analysis pipelines. This work is further motivated by the success of the BioCreative I and II evaluation challenges as well as the increasing availability of resources and use of standard public corpora such as AIMed, BioInfer, GENIA, PennBioIE and the resources created for BioCreative.

Considerable progress has been made in foundational tasks such as named entity recognition, where the BioCreative II task organizers demonstrated the feasibility of recognition with mean precision and recall in excess of 90%. However, as the BioCreative evaluation also demonstrated, even after ten years of study, significant challenges still remain in the key task of protein-protein interaction extraction, a problem that maintains its relevance to the community and toward which these proceedings also hold a number of contributions

SMBM 2008, hosted by Turku Centre for Computer Science (TUCS) in Turku, Finland is the third in the series of International Symposia on Semantic Mining in Biomedicine, following SMBM 2005 at EMBL-EBI in the U.K. and SMBM 2006 at Friedrich-Schiller University in Jena, Germany. The event aims to bring together different communities: researchers from text and data mining in biomedicine, medical-, bio- and cheminformaticians, and researchers from biomedical ontology design and engineering. Further, we strongly encourage constructive dialogue between academia and industry, and gratefully acknowledge the support of our academic and industry sponsors.

This year, we received 38 submissions in three categories: full papers, short papers, and Work in Progress Proposals. The overall quality of the submissions was very good and, consequently, the selection process became competitive: we accepted about one half of the manuscripts in each of these categories. Additionally, a number of authors of full papers were invited to resubmit a short version of their manuscript. In total, 15 full papers, 10 short and two Work in Progress Proposals are included in the proceedings at hand. We additionally include abstracts of two invited talks.

Dear reader, we hope that you will enjoy reading these proceedings and find them interesting and inspiring to your work.

August 2008

Organization

SMBM 2008 was organized by Turku Centre for Computer Science (TUCS) with support from EMBL-EBI, University of Turku, and Åbo Akademi.

Programme Chairs

Dietrich Rebholz-Schuhmann	EMBL-EBI, Hinxton, UK
Tapio Salakoski	TUCS, Turku, Finland

Local organization chair

Sampo Pyysalo	TUCS, Turku, Finland

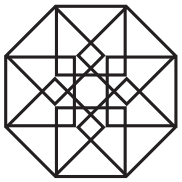
Programme Committee

Sophia Ananiadou	University of Manchester and NaCTeM, UK
Christopher J. O. Baker	I2R, Singapore
Olivier Bodenreider	National Library of Medicine, USA
Anita Burgun	University of Rennes, France
Kevin Cohen	University of Colorado, USA
Nigel Collier	National Institute of Informatics, Japan
Juliane Fluck	Fraunhofer SCAI, Germany
Carol Friedman	University of Columbia, USA
Filip Ginter	University of Turku, Finland
Udo Hahn	University of Jena, Germany
Lynette Hirschman	MITRE, USA
Su Jian	I2R, Singapore
Jin-Dong Kim	University of Tokyo, Japan
Satoshi Kobayashi	University of Electro-Communications, Japan
Michael Krauthammer	Yale University School of Medicine, USA
Patrick Lambrix	Linköping University, Sweden
Adeline Nazarenko	Université Paris-Nord, France
See-Kiong Ng	NTU and I2R, Singapore
Jong C. Park	KAIST, South Korea
Martin Romacker	Novartis Pharma AG, Switzerland
Gerold Schneider	University of Zurich, Switzerland
Stefan Schulz	Freiburg University Hospital, Germany
Hagit Shatkay	Queen's University, Canada
Jun'ichi Tsujii	University of Tokyo, Japan and University of Manchester, UK
Alfonso Valencia	CNIO, Spain
Simo Vihjanen	Lingsoft Inc., Finland
Limsoon Wong	National University of Singapore

Local Organizers

Hanna Suominen
Filip Ginter
Anna Santanen

SMBM'08 has received generous financial support from a number of institutions and private companies. Their contribution is hereby acknowledged.



The Federation of Finnish Learned Societies

Turku University Foundation

Åbo Akademi Foundation

turku

SCIENCE TURKU PARK

Lingsoft® LANGUAGE SOLUTIONS

Invited talks

Full papers

Short papers

Work in progress proposals

Invited talks

Text Mining Methods as Computational Biology Tools

Alfonso Valencia
Structural Biology and BioComputing Programme
Spanish National Cancer Research Centre (CNIO)
valencia@cnio.es

During the last few years many new Information Extraction and Text Mining methods have been developed and many of them are accessible on the web. Still, we do not have many examples of their integration with those commonly applied to biological problems in Genomics and Systems Biology, despite the current general recognition of the need to use extensively and systematically the information directly extracted from textual sources.

My group has been working in integrating Text Mining approaches in large-scale projects, together with other complementary experimental and bioinformatics methods. In particular in the ENFIN project we have developed new approaches to collect information on proteins interacting with proteins known to form part of the human spindle body complex and to systematically score them by the likelihood of their implication in the formation of the spindle. The predictions of this Text Mining method, combined with those of a collection of other methods based on sequence and structure analysis, have been followed up by detailed experimental verification including in situ localization assays and iRNA screenings. Furthermore, we have developed a Text Mining system to assist human experts in the annotation of spindle related proteins that have allowed us to generate a large collection of validated proteins and text pieces. This new system is now being use to train and test the sequence / structure based prediction methods.

For these, and other, applications of Text Mining it is crucial to have an accurate estimation of the capacity of the current systems. The BioCreative II challenge organized by CNIO, MITRE and NCBI in collaboration with the MINT and INTACT databases (http://biocreative.sourceforge.net, Genome Biology, August 2008 Special Issue) provides such an overview. BioCreative II was organized in two tasks:

1. gene name identification and normalization, where many systems were able to achieve a consistent 80% balanced precision / recall.

2. protein interaction detection, which was divided into four sub-tasks:
   (a) ranking of publications by their relevance on experimental determination of protein interactions
   (b) detection of protein interaction partners in text
   (c) detection of key sentences describing protein interactions and
   (d) detection of the experimental technique used to determine the interactions.

The results were good in the categories of publication raking, detection of experimental methods, and highlighting of relevant sentences, while they pointed to persistent problems in the correct normalization of gene/protein names. It is interesting to notice that the typical performance of the best Text Mining methods is not very different from that of many standard bioinformatics methods, for example structure prediction and protein docking methods. Furthermore, BioCreative has channeled the collaboration of several teams for the creation of the first Text Mining meta-server (The BioCreative Meta-server, Leitner et al., Genome Biology 2008 BioCreative special issue). We are now working in the preparation of BioCreative III, with particular focus in fostering the creation of Text Mining systems that can be integrated in Genome analysis pipelines.

Natural Language Processing in the Medical and Biological Domains: a Parallel Perspective

Pierre Zweigenbaum
LIMSI - CNRS
BP 133, F-91403 Orsay Cedex, France
http://www.limsi.fr/~pz/
pz@limsi.fr

Abstract

Natural Language Processing (NLP) has been active in the medical domain for more than thirty years, with pioneering projects such as the Linguistic String Project. 'BioNLP', the application of Natural Language Processing methods to the analysis of the biological literature in the genomics era, has undergone a fast development in little over ten years.[1] It rapidly attracted Medical NLP and Computational Linguistics researchers, especially through challenges and evaluation initiatives. We examine here to which extent medical NLP prepared the ground for BioNLP. Conversely, we study the ways BioNLP influenced the practice of medical NLP.

1 Medical NLP: Specificities and Contributions to BioNLP

A growing community of researchers applies NLP to the medical domain and develops new methods for that purpose. Medical NLP has seen important breakthroughs, such as routine, machine analysis of clinical reports (MedLEE), but it is probably fair to say that it has had until now only a moderate direct impact on clinical applications. It has been mostly concerned with the clinical domain (clinical notes, etc.), but also with the analysis of the scientific literature (MEDLINE titles and abstracts).

1.1 Contributions

We wish to stress nevertheless that medical NLP prepared the ground for BioNLP by providing resources and tools which could be reused in that domain. A large effort was spent on the creation of lexical (*e.g.* the UMLS Specialist Lexicon) and unified terminological resources (*e.g.* the terminologies which can be found in the UMLS Metathesaurus). These resources are used for instance in automatic term recognition, *e.g.* in MetaMap, a widely used component in BioNLP systems. Medical ontologies have also seen a continuous stream of work since GALEN (*e.g.* Foundational Model of Anatomy, SNOMED CT), whose methods could help the design of the Gene Ontology. Indexing and Information Retrieval from the medical literature (*e.g.* SAPHIRE, MTI) and from health records (*e.g.* ICD and SNOMED coding) are long-standing research topics. They aim at concept-based indexing (*e.g.* MetaMap), a task similar to the gene normalization task of the BioCreAtIvE challenges. Text analysis was an early target of Medical NLP. It led to successful systems which were then applied to the biomedical domain (*e.g.* GENIES[2], SEMREP[3]), porting from one sublanguage to the other (Friedman et al., 2002). Finally, literature-based discovery as started by Swanson (inasmuch as it uses NLP) was applied to medical problems long before it was geared towards biomedical knowledge.

1.2 Specificities

The medical domain has specific features which bear consequences on medical NLP research. First, the requirement for *privacy* of clinical records has had a strong impact on clinical NLP. It prevents researchers from sharing text corpora (NLP based on the medical literature does not have this limitation). Deidentification methods

[1] For an introduction, see *e.g.* (Ananiadou and McNaught, 2006).

[2] http://www.cat.columbia.edu/genies.htm

[3] http://skr.nlm.nih.gov/papers/

have been investigated to overcome this barrier, but human review is generally still needed. Second, *localisation* of clinical records and associated functionality is necessary. Clinical records must use the language of the user, which entails a need to develop resources for each natural language (terminologies and ontologies, being designed as concept representations abstracted from natural languages, are an exception: they can be shared across languages). This has created strong constraints on the sharing of resources and tools, which led to the dispersion of concrete efforts in medical NLP. Besides, medical NLP has tackled *several specific sublanguages* beyond that of the biomedical literature: that of clinical reports, often with short phrases and terse style, and more recently those of practice guidelines and patient-oriented documents, with constraints of readability and understandability. Finally, it has addressed *diverse user needs*, mainly those of a variety of health care professionals and administrative staff in hospitals (clinical documents) and those of researchers (articles). This dispersed market segmentation also tends to disperse research.

2 BioNLP: Specificities and Contributions to Medical NLP

2.1 Contributions

BioNLP promotes a dynamic, shared way of conducting research within a community. This can be seen in the organization of challenges (*e.g.* TREC Genomics or BioCreAtIvE). These depend on shared annotated corpora (*e.g.* GENIA[4]), a key component in such initiatives. In contrast, medical NLP researchers have had to overcome the above-mentioned strong limitations on the development of clinical corpora to launch such challenges. This has recently started with the i2b2 de-identification challenge (2006)[5] and the Cincinnati Medical Center ICD-9-CM coding challenge (2007)[6], and the AMIA NLP working group is striving to foster this process. Sharing in BioNLP also applies to information extraction pipelines, *e.g.* LingPipe or the JULIE tools.

BioNLP has had a faster and wider attraction of 'mainstream' Computational Linguistics researchers. More attractive funding of the genomics domain probably played a role here, but the availability of corpora and resources and the organisation of challenges were certainly very important factors too. Directly usable training and testing corpora have also allowed many researchers to test Machine Learning methods (*e.g.* to recognize gene mentions) with only minimal investment in the specificities of the domain.

2.2 Specificities

BioNLP has the great advantage of working mostly on common input documents: the biomedical literature. There is no need for privacy here (but access rights are enforced on the majority of full-text articles), and documents are written in one language: 'bio English'. The biomedical sublanguage inherits that of scientific, experimental literature; it also has specific components, *e.g.* for gene and protein names and interactions. The development of lexical, terminological and ontological resources for these components has therefore been the subject of much work in BioNLP. The emphasis of BioNLP is on text mining, and it has more focused targets, namely, researchers and database curators. This may form a more homogeneous user base than that of medical NLP.

3 Conclusion

Based on the above comparison, hypotheses can be formulated to explain differences between the attractivity and development speed of medical- and BioNLP. Not yet mentioned is the intrinsic scientific attractivity of the biomedical domain, with a promise of more fundamental outcomes. Funding is indeed an important factor too. Nevertheless, we believe the importance of enabling factors must be stressed: a shared input language facilitates shared resources and tools, no requirement for privacy enables shared corpora and the organisation of challenges, which have been a driving force in BioNLP.

[4] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/

[5] https://www.i2b2.org/NLP/Workshop.php

[6] http://www.computationalmedicine.org/challenge/

References

Sophia Ananiadou and John McNaught, editors. 2006. *Text mining for biology and biomedicine*. Artech House Publishers.

Carol Friedman, Pauline Kra, and Andrei Rzhetsky. 2002. Two biomedical sublanguages: a description based on the theories of Zellig Harris. *J Biomed Inform*, 35(4):222–235.

Full papers

Towards Semantic Annotation of Bioinformatics Services: Building a Controlled Vocabulary

Hammad Afzal, Robert Stevens, Goran Nenadic
School of Computer Science, University of Manchester, Manchester, UK
{H.Afzal@postgrad., R.Stevens@, G.Nenadic@}manchester.ac.uk

Abstract

Most bio text-mining efforts so far have focused on identification of biological, molecular and chemical entities from the literature to support knowledge acquisition and discovery in the life sciences. There are also a growing number of bioinformatics services and tools available. This raises the challenging problem of semi-automated annotation, documentation and discovery of services suitable for a specific data analysis and/or integration into workflows. The first step in this process would be to build a controlled vocabulary to describe bioinformatics services, which can then be used for service retrieval and discovery. In this paper we present a methodology that combines lexical and contextual profiles of candidate terms to suggest terms for the bioinformatics vocabulary. The method achieved an estimated precision in the range 70-90% with recall between 20 and 90%. After processing the whole of BMC Bioinformatics, almost 80% of the top 300 terms were deemed as conceptual terms relevant for describing the major concepts in bioinformatics. In addition to this, the method has also extracted a number of service and tool names. The controlled vocabulary is freely available at: http://gnode1.mib.man.ac.uk/bioinf/CV.

1 Introduction

Along with the huge amount of experimental data, both raw and curated, and together with the literature being published in the biomedical domain, various bioinformatics data sources and tools have exposed programmatic interfaces as services. Resource sharing has already been established as a common policy within the community, and many groups have dedicated significant efforts to organise both internal and public repositories of bioinformatics tools, typically classifying them in broad categories (e.g. EBI Web services[1] are organised into data retrieval, analytics, similarity searches, multiple alignment, literature processing, etc.). Several projects and initiatives (e.g. myGrid[2] and myExperiment[3]) are annotating functional capabilities and semantically describing resources in a way which would make them discoverable and usable both by bioinformaticians and machines. Service descriptions typically include both textual explanations and ontological annotations. For example, EBI's *emma* service[4] is represented by the following (textual) description in the myGrid repository:[5]

> *Performs a multiple alignment of nucleic acid or protein sequences using ClustalW program*

along with a set of myGrid ontology[6] tags describing its operation (*multiple local aligning*), type (*Soaplab service*), parameters (including name, semantic type (e.g. *biological sequence*) and format (e.g. *single sequence format*), etc.).

Currently, most of the frameworks cataloguing bioinformatics services and workflows (e.g. myGrid/Taverna (Oinn et al, 2007)) describe resources manually, which – like any curation task – requires a lot of time and effort. As the number of services is increasing, manual annotation is becoming a bottleneck for discovering and using relevant services and tools (Cannata et al, 2005). Therefore, (semi)automatic methodologies to describe services are becoming inevitable, including automatic extraction of functional descriptions of services from available documents (articles, blogs, documentation, user manuals, etc.). Furthermore, since the domain is extremely dynamic, controlled vocabularies and/or ontologies that are (or can be) used for annotations need to be regularly updated and adjusted to include emerging methods, functionalities, data formats, etc. For example, the myGrid ontology

[1] http://www.ebi.ac.uk/Tools/webservices/

[2] http://www.mygrid.org.uk/

[3] http://www.myexperiment.org/

[4] http://www.ebi.ac.uk/soaplab/emboss4/services/alignment_multiple.emma

[5] http://www.mygrid.org.uk/feta/mygrid/descriptions/Soaplab_EBI/alignment_multiple/emma.xml

[6] http://www.mygrid.ac.uk/ontology

(Wolstencroft et al, 2007) contains around 440 bioinformatics terms that are currently used to describe services; still, for many potentially useful services, there may not be a set of adequate ontological terms or keywords for their description. For example, it would be difficult to precisely describe *GeneSom* service (Yan, 2002) for clustering-based microarray data analysis, as term *clustering* is not included in the current set of the ontology terms (the closest related term would be *grouping*).

In this paper we present a methodology and results in building a controlled vocabulary (CV) of bioinformatics terms that can be used for semantic annotation and description of services. By CV, we mean a set of key terms that are used to convey relevant information in a given domain or task (Kageura and Umino, 1996). Our main hypothesis is that new potential descriptors are likely to appear in documents that report on service design or utilisation. Therefore, our method for identification of terms related to bioinformatics services is based on processing full text articles from relevant journals. We have combined an automatic term recognition technique with a term classification approach based on lexical and contextual properties of candidate terms. Since not all terms that appear in a given corpus are of interest for a given task (e.g. specific protein/gene names, drugs, etc. may not be of interest to service descriptions), the method aims to filter out candidate terms that are not relevant for the task. The results obtained are very encouraging, showing that 70-90% of terms obtained are relevant for service descriptions, making the CV generation a first step towards facilitating automated annotation of services.

The paper is organised as follows. In Section 2 we present the overall methodology. The results and discussions are presented in sections 3 and 4 respectively, while related work is examined in Section 5. Finally, Section 6 concludes the paper and gives an outline of topics for future work.

2 Methodology

We have designed the following general methodology (see Figure 1): we start with the candidate term recognition process from a corpus and apply a classification method that rearranges the candidate terms according to their relevance to the task and/or domain of interest (in our case bioinformatics tools/services**)**. Term classification is based on a hybrid approach combining terms' lexical and contextual properties, represented as term profiles. Task/domain relevance is then assessed by comparing profiles of candidate terms with profiles of seed terms and ontological concepts that portray the task/domain. These steps are described below in detail.

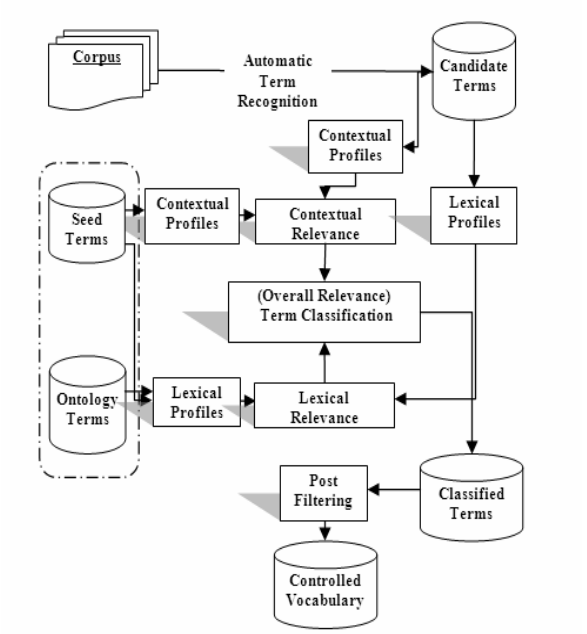


Figure 1: System architecture

2.1 Collecting candidate terms

To support the task, we have collected a corpus of full text articles from a bioinformatics journal. The corpus was processed by an Automatic Term Recognition (ATR**)** service (TerMine[7], based on the C-value method (Frantzi et al, 2000)), in order to obtain the candidate terms to be considered for the CV. The C-value method is, however, a generic ATR approach that considers only statistical information (frequency of occurrence, string nestedness, etc) and recognises terms that are relevant for the whole collection, irrespective of sub-domains/tasks of interest. Therefore, the candidate terms would typically include gene and protein names, drugs, organisms, chemical terms, various procedures, tools, etc. Our aim is to identify only terms related to bioinformatics by assessing correlation between the candidate terms and a set of pre-prepared concepts representing the task/domain of interest.

2.2 Knowledge resources

To represent the domain, we have created a knowledge base that comprises two resources: a list of seed terms and a list of ontological terms. Both resources are used to provide the lexical profile of the domain, with the seed terms also used to "illustrate" textual behaviour in docu-

[7] http://www.nactem.ac.uk/software/termine/

ments (i.e. pragmatics) of the domain terms, providing positive "use cases". Obviously, ontological terms may not appear in the literature since they describe concepts and are used for domain modelling. For the bioinformatics CV task, the seed terms (ST) have been collected from existing Web service descriptions provided by various sources (e.g. EBI Web services) and from the relevant literature cited at the myGrid website[8]. These terms have been collected automatically using TerMine and then manually pruned on the basis of their relevance to our domain. A total of 250 terms have been identified: these are "real" terms used for service descriptions in the literature. Ontological terms (OT) are extracted from 440 concepts of the bioinformatics ontology prepared by the myGrid team. The ontology includes informatics concepts (the key concepts of data, data structures, databases and metadata); bioinformatics concepts (domain-specific data sources e.g. model organism sequencing databases, and domain-specific algorithms for searching and analysing data e.g. the sequence alignment algorithm); molecular biology concepts (higher level concepts used to describe bioinformatics data types, used as inputs and outputs in services e.g. protein sequence, nucleic acid sequence); task concepts (generic tasks a service operation can perform e.g. retrieving, displaying and aligning).

For each of the seed and ontological terms, we have generated (as explained below) lexical profiles that will be used to identify potential bioinformatics terms. For the seed terms only, we have also generated contextual profiles to provide a case-base with typical contexts in which the seed terms have appeared.

2.3 Term profiles

The main idea behind our term classification process is to measure the degree of similarity between candidate terms and the known bioinformatics terms by comparing their lexical (constituents) and contextual (pragmatics) profiles.

Lexical profiles. Each candidate term is assigned a lexical profile, represented by all possible left-linear combinations of the word-level substrings present in a term (Nenadic and Ananiadou, 2006). For example, the lexical profile of the term *protein sequence alignment* is the following set: {*protein, sequence, alignment, protein sequence, sequence alignment, protein sequence alignment*}. These profiles are then compared (as sets) to the profiles of the seed and ontological terms. The hypothesis here is that – since scientific sublanguages are characterised by words and their collocations which appear more frequently in a given domain (Kittredge, 1982) – we can use lexical correlations to suggest potential candidates.

We have employed two different approaches: comparing a candidate term profile using an "average" bioinformatics seed/ontology term (LR_1, formula (1) below) and finding the best match (LR_2, formula (2)). In both cases we use a Dice-like coefficient to measure the lexical relevance. If LP(t) represents the lexical profile of a term t, and LP(s_i) and LP(o_i) lexical profiles of a seed and ontological term respectively, then lexical relevance of term t is calculated as follows:

$$\text{LR\_1(t)} = \frac{1}{2n}\sum_{i=1}^{n}\left(\frac{2(\text{LP(t)} \cap \text{LP}_i(s_i))}{|\text{LP(t)}| + |\text{LP}_i(s_i)|}\right) + \frac{1}{2m}\sum_{j=1}^{m}\left(\frac{2(\text{LP(t)} \cap \text{LP}_j(o_j))}{|\text{LP(t)}| + |\text{LP}_j(o_j)|}\right) \quad (1)$$

$$\text{LR\_2(t)} = \max_{\substack{i=1 \text{ to } n \\ j=1 \text{ to } m}}\left(2\frac{\text{LP(t)} \cap \text{LP}_i(s_i)}{|\text{LP(t)}| + |\text{LP}_i(s_i)|}, 2\frac{\text{LP(t)} \cap \text{LP}_j(o_j)}{|\text{LP(t)}| + |\text{LP}_j(o_j)|}\right) \quad (2)$$

Here, n and m represent the total number of seed terms and ontological terms respectively. In case of LR_1, we estimate lexical relevance on the basis of its relative similarity to the whole domain, whereas, in case of LR_2, we focus on maximal similarity to a seed or ontological term.

Contextual profiles. Target terms may have no lexical resemblance to the seed/ontological terms. For example, *fisheye* is a name of a tool that cannot be identified as a relevant bioinformatics term based only on its lexical properties. We therefore consider contexts (namely sentences) in which candidate terms occur in order to profile their behaviour using co-occurring nouns and verbs, as well as lexico-syntactic patterns in which candidate terms occur. A contextual profile of each term comprises its noun sub-profile, verb sub-profile and context pattern sub-profile. Similarly, contextual profiles of the seed terms are built using the literature from which they have been extracted.

Contextual elements are identified using a POS tagger and lemmatiser (the Genia tagger[9] was used), parser (Stanford parser[10]) and the TerMine service. As a result of pre-processing and filtering non-content bearing units (including modals and adverbs), each sentence is repre-

[8] http://www.mygrid.org.uk/wiki/Mygrid/BiologicalWebServices

[9] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/tagger/

[10] http://nlp.stanford.edu/software/lex-parser.shtml

sented as a stream of lexico-syntactic (noun phrases, verb phrases, prepositions) and terminological units, with their relative positional information with respect to the candidate term. In our experiments, we have used two types of unit representation (see Table 1 for an example). In the first type, noun phrases and terms are represented by their class only (as NP and Term respectively), whereas verb phrases and prepositions are represented by their lemmas. We have not generalised verbs and prepositions since they are expected to carry useful information for classification of candidate terms (Spasic and Ananiadou 2004). The second type of pattern contains lemmas for all units, including NPs and terms. A pattern profile is then represented by left (LP**)** and right (RP**)** patterns, which represent units appearing on the left and right side of the candidate term in a sentence respectively.

Verb profile	*produce*
Noun profile	*Genscan, program, list, transcript*
LPs	Term, *produce*, NP, *of* *Genscan program, produce, a list, of*
RPs	*of*, NP *of, predicted transcripts*

Table 1: An example of contextual profiles of the term *nucleotide FASTA*, originated from the following sentence: *The Genscan program can produce a list of nucleotide FASTAs of predicted transcripts.* The first line in LPs and RPs rows represents the first pattern type (lemmas for verbs and prepositions only), while the other represents the second type (lemmas for all constituents).

Contextual profiles are used to measure contextual relevance (CR) of each candidate term by comparing them to the contextual profiles of the seed terms. Similarly to lexical relevance, we have used two formulae comparing a candidate term profile to the average seed term, and to the most similar one:

$$\text{CRN\_1(t)} = \frac{1}{n}\sum_{i=1}^{n}\left(2\frac{\text{CPN(t)} \cap \text{CPN}_i(s_i)}{|\text{CPN(t)}| + |\text{CPN}_i(s_i)|}\right) \quad (3)$$

$$\text{CRN\_2(t)} = \max_{s_i \in ST}\left(2\frac{\text{CPN(t)} \cap \text{CPN}(s_i)}{|\text{CPN(t)}| + |\text{CPN}(s_i)|}\right) \quad (4)$$

Here, CPN(*x*) represents a contextual noun profile of (a candidate or seed) term *x*. Relevance measures using verbs (CRV) and patterns (CRP) are calculated similarly. In addition to these term-term comparisons, we also consider *aggregate* contextual seed profiles comprising features (i.e. nouns, verbs, LPs, RPs) appearing in context of any seed term. Using these values, the aggregate contextual (noun) relevance is calculated as

$$\text{CRN\_3(t)} = 2\frac{\text{CPN(t)} \cap \text{CPN\_A}(ST)}{|\text{CPN(t)}| + |\text{CPN\_A}(ST)|} \quad (5)$$

where CPN_A(*ST*) is the aggregate noun profile of the seed terms. Similar approaches are followed for verb and pattern profiles.

2.4 Building the controlled vocabulary

As described before, our main aim is to provide a methodology to automatically build a terminological resource containing terms that are similar lexically and pragmatically to a given set of terms from the knowledge resources. Our approach is based on combining the four types of profile similarities to estimate the overall relevance **(**OR**)** of a candidate term:

$$\text{OR(t)} = \theta\, LR(t) + \alpha \cdot CRN(t) + \beta \cdot CRV(t) + \gamma \cdot CRP(t) \quad (6)$$

where LR(*t*), CRN(*t*), CRV(*t*) and CRP(*t*) represent relevance of term *t* based on lexical, contextual nouns, contextual verbs and contextual pattern profiles respectively. The parameters α, β, γ and θ can be used to assign different weights to the profiles' contributions. By applying term weighting, we can obtain a list of candidate terms with high OR, and extract/classify terms with OR above a certain threshold as relevant and consider them for the CV building. The threshold value and term post-filtering can be varied according to the user's requirement for precision and recall.

3 Experiments and Results

To assess the suggested method, we have performed three experiments. First, we have evaluated the performance (precision/recall) of term classification on a subset of documents. Then, we have evaluated the top 300 terms extracted by the system with regard to precision, and finally estimated the recall as compared to the myGrid bioinformatics ontology.

The knowledge resources used in the experiments are as follows. We used 250 seed terms and 440 ontological terms, for which lexical and contextual (ST only) profiles have been generated. The corpus from which we collected candidate terms consisted of 2120 full text open-access articles from *BMC Bioinformatics*[11] (published before March 2008). Full text is essential for this

[11] http://www.biomedcentral.com/bmcbioinformatics/

task, firstly because we expect to find many of the candidate terms in the methods section, and, secondly, as it is more likely to find detailed contexts for term classification in full text documents rather than in abstracts only. After applying the C-value method on the corpus, we have collected almost 100,000 candidate terms (see later Table 4 for detailed statistics) and generated their lexical and contextual profiles.

We used 135 additional bioinformatics terms manually extracted from the service describing literature cited on the myGrid website for tuning the system parameters (cf. formula (6)). A genetic algorithm iterative procedure given in (Spasic et al, 2004) has been performed to learn the parameters to optimise the results on the tuning terms so that the maximal number of the tuning terms ends up in the top 10% of the suggested candidate terms. We randomly varied the values of parameters through 1000 iterations, providing that $\alpha+\beta+\gamma+\theta=1$. In each optimisation cycle we have considered all individual profiles (e.g. LR_1, CRN_2, etc.) or their combinations (e.g. LR_1 & CRN_3 & CRV_2 & CRP_3) so to find the best performing values of the parameters. While the max-based lexical similarity (LR_2) was better than the average-based LR_1, there were no significant differences between various contextual formulae. Still, the best overall performance on the tuning terms was when we combined CRN_1, CRV_3, CRP_2 and LR_2 with the following parameter values: α = 0.355, β= 0.158, γ = 0.02 and θ = 0.462 (used as the default parameters further on).

Experiment 1: term classification performance. In order to estimate the precision and recall of the term classification part, we have randomly selected a subset of five documents, in which 375 terms appear (automatically recognised by TerMine). These have been manually classified by a domain expert as relevant/irrelevant. We have then evaluated the system performance (using the usual metrics for precision, recall and F-measure) on this set. The best performing individual metrics (LR_2, CRN_1, CRV_3 and CRP_2) are summarised in Table 2. Table 3 summarises the performance of three combined profiles with the best performance. The best results were obtained when CRN_1, CRV_3, CRP_2 and LR_2 were combined, with precision in the 70% range and recall in the 90% range (F-measure in the 80% range). Figure 2 summarises the results for the best performing metrics.

	LR_2	CRN_1	CRV_3	CRP_2
Precision	69.1	63.4	71.2	**80.6**
Recall	83.3	77.0	62.7	19.8
F-measure	75.5	69.5	66.7	31.8

Table 2: The performance of the best individual metrics on the test set (375 terms).

	Comb1	Comb2	Comb3
Precision	68.2	67.9	67.1
Recall	**92.1**	84.1	**92.1**
F-Measure	**78.4**	75.2	77.2

Table 3: The performance of combined metrics on the test set (375 terms). [Comb1 = CRN_1, CRV_3, CRP_2 and LR_2, with the default parameters; Comb2 using only CRN-1 and LR2 with α = 0.298 and θ = 0.702; Comb3 using CRN_1 and CRV-3 with β= 0.258 and θ = 0.742].

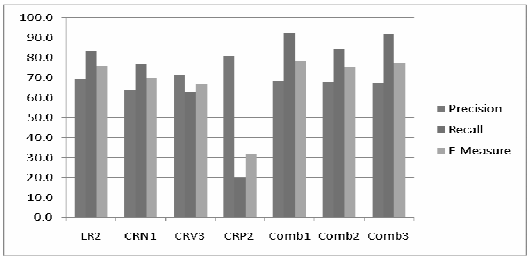


Figure 2: The performance of individual metrics

Experiment 2: the controlled vocabulary precision. Two domain experts evaluated the top 300 terms as suggested by the system. The results (see Fig. 3) have showed that the top 100 terms were highly relevant, with 93% of terms deemed suitable to make a direct entry into the bioinformatics CV. The precision for the top 200 terms fell to 83% and to 79% for the top 300 terms.

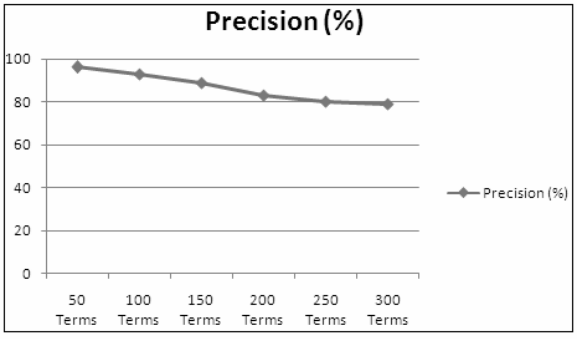


Figure 3: Precision of the top 300 CV terms

Experiment 3: "reconstructing" the myGrid bioinformatics ontology. In addition to estimating recall for the term classification task, we investigated to what extent the system could reconstruct the myGrid bioinformatics ontology. The experiments have shown that even 45 terms (10% of the myGrid ontology) appeared in the first 100 terms, totalling to 59 (13.4%) for the first 300 terms (see Figure 4). We have also found that the total of 20% of the suggested top 300 terms fully matched the corresponding myGrid concepts.

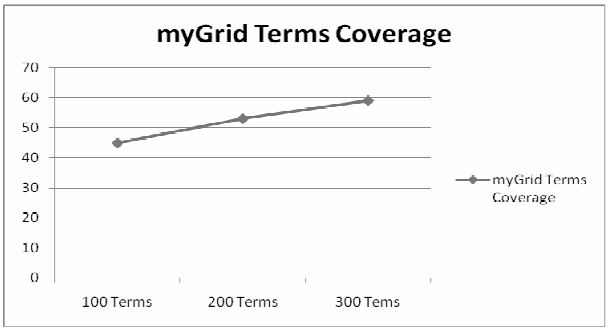


Figure 4: The number of myGrid terms recovered

4 Discussion

We have presented a generic methodology to automatically build and expand a controlled vocabulary in a domain of interest using literature mining. For this purpose, we employed a term classification approach that combines lexical and contextual properties of candidate terms and compares them to seed entities.

In the experiment 1 of the bioinformatics CV building task, the best individual performance (the best F-measure) was observed in the case of lexical relevance (see Figure 2 and Table 2). In addition to lexical properties of candidate terms (that typically give precise results but fail to identify some relevant terms), we also consider contextual profiles. Table 4 shows the number of terms being recognised using various metrics. Most of the top suggested terms made into the CV based on their lexical profiles, but there were still terms that were only contextually similar to the seed terms (e.g. terms such as *statistical approach* or *SVM classifier*; or *biological text* (e.g. as an input concept)).

Overall, in the case of lexico-syntactic patterns, the results show very good precision (see Table 2, last column). The reason is that the patterns originated from the seed terms were able to model the pragmatics of bioinformatics terms. Since the number of seed terms and their contexts were limitted (250 terms and 1034 contexts), this has resulted in an accetably low recall. In the experiments we have varied the representation of patterns (considering the generic classes of neighbouring units and varying the length of the window of neighbouring words), and the best balance between precision and recall was obtained using three neighbouring units.

When lexical and contextual profiles were combined, a significant increase in recall was observed as compared to individual metrics (see Table 3 – recall of 92.1% as compared to 83.3% for lexical and 19.8% for contextual), with no significant drop in precision (if at all), resulting in the overall improved F-measure.

We have also varied the seed terms used by swapping 100 out of 250 terms with new terms collected using the same methodology from the service description literature. However, the results were similar, showing small variations of 1-2% in precision and recall.

Total number of candidate terms collected using ATR	98,986
Number of terms classified using lexical similarity	61,977
Number of terms classified using contextual nouns	84,412
Number of terms classified using contextual verbs	64,477
Number of terms classified using contextual patterns	17,638

Table 4: The number of terms suggested from the BMC corpus using different similarity metrics

The evaluation of the top 300 candidate terms revealed that there were three term types suggested. The first type relates to terms that refer to a generic concept related to bioinformatics services and can make a direct entry into the CV (direct true positives). More than half of all terms are in this category. The second category contains terms that would need slight modification before becoming part of the CV. For example, this type includes units that begin with a generic or non-specific modifier (e.g. *user friendly* in *user friendly Gpcr oligomerization knowledge base*), or wrongly identified terminological head (e.g. *compromise* in *tab delimited text file compromise*). We have applied simple rules to fix these issues, improving the number of (direct) entries by more than 11%. The third type contains names that refer to toolkits, workbench platforms, databases etc. (e.g. *protein visualization tool RASMOL*, *myGrid Taverna workbench*, etc.) They do not refer to generic concepts and thus

are not direct entries to the CV, but are of interest for the service discovery process. Note that such terms were also included in the seed term set, so their contextual profiles were used as positive use cases. Overall, adding these terms improved the total precision to 79.3%.

The experiments with the myGrid ontology (experiment 3) were interesting in the sense that the suggested method was promising in both reconstructing the terms from an ontology (reasonable recall), but also in identifying new potential entries or synonyms that could be used (e.g. terms such as *life science identifier* (*LSI*), *systems biology mark up language* (*SMBL*), etc. have been suggested).

5 Related work

There have been several approaches to semi-automated building of controlled vocabularies and ontologies from literature (Grefenstette 1994). For example, Spasic et al. (2008) present a methodology for development of CVs for metabolomics. They employed an automatic term recognition method to identify candidate terms from a corpus and then filtered relevant terms on the basis of their semantic association to a set of manually chosen relevant concepts. In this case, the UMLS[12] was used as a (static) semantic model to identify properties to which target terms should conform.

Sabou et al. (2005) present an automatic method that learns domain ontologies for Web service descriptions from textual information attached to Web services. They annotated a corpus with linguistic information and then performed syntactic parsing and employed a set of syntactic patterns to identify and extract information from the corpus. The patterns are focused on domain concepts, their functionalities (verbs associated with concepts) and inter-relationships between concepts (via prepositions). This extracted information is then transformed into a structured ontology.

Automatic term classification is also related to our work, in particular for different biological entities (e.g. gene and protein mentions (Yeh et al, 2005)). The reported methodologies include keyword-driven approaches, where biomedical terms containing functional words such as *receptor*, *factor* or *radical* are used to assign term categories. These functional words may not always be discriminative, and determination of term class is not possible merely by comparing the functional words, which may lead to the ambiguity in term classification (Krauthammer and Nenadic, 2004). Statistical and machine learning approaches are also used (Collier et al, 2000; Lui and Friedman 2003). For example, an approach for disambiguation between proteins, genes, and mRNAs using different machine learning techniques (naïve Bayesian classification, decision trees and inductive learning) was reported by Hatzivassiloglou et al. (2001). They used different features for classification including words that appeared near a term, positional, morphological, distributional and shallow syntactic information about terms and reported an overall accuracy between 69.4% and 85% for a two-way classification task (gene/protein) and between 65.9% and 78.1% for a three-way classification task (gene/protein/ mRNA).

Apart from using morphological, lexical and syntactical properties of a term, key features from the context of a term occurrence can also be employed to determine the class of that term. For example, Al-Mubaid (2006) used mutual information and χ^2 as feature selection techniques to identify the best features from term contexts to build a term classifier. Similarly, Spasic et al. (2005) combined machine learning and domain knowledge (the UMLS thesaurus) to design a case-based reasoning system for term classification based on context alignment (using the edit distance similarity between syntactic and semantic constituents). In their previous work, Spasic and Ananiadou (2004) also used automatically learnt verbal preferences to support classification of biomedical terms.

6 Conclusions

Most bio text-mining approaches so far have focused on identification of biological and molecular entities from the literature to support knowledge acquisition and discovery in the life sciences (Jensen et al, 2006), with very few attempts to characterise the bioinformatics sublanguage and the terminology used to present technologies, experimental procedures and methodologies. In this paper we have focused on a controlled vocabulary that can be used to semantically annotate bioinformatics services and tools.

We have presented a term classification driven methodology to automatically build a CV for a domain represented with a set of seed terms and (optionally) a set of ontological descriptions. The methodology integrates lexical and contextual profiles of candidate terms, and compares them to the available resources. In the lexical ap-

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

proach, we quantify the degree of sharing of constituents between candidate terms and the seed and ontology units. In the context-based profiling, we model textual behaviour of terms, using co-occurring nouns and verbs, or describing contexts using contextual patterns. While the ontological concepts are used only to capture the lexical dimension of the domain conceptual space, the seed terms are also used to describe pragmatics of the given domain through a set of "known" use cases.

The results of the methodology applied to the bioinformatics domain revealed that the approach is useful for a rapid creation of a CV. We have processed all of the BMC Bioinformatics articles, with the estimated best precision of around 70% and recall of 90%. The precision for the top 100 suggested terms was 93%.

The CV generation can be viewed as a first step towards facilitating the automation of the service description process by not only aiding in the provision of baseline terminologies, but also by providing a useful lexical resource that can then be utilised for other NLP tasks like information retrieval, named entity recognition and information extraction in the bioinformatics domain. Future work will include incremental learning of terms and identification of patterns that are relevant for service descriptions, as well as more detailed identification of roles that bioinformatics terms may have in a given context (e.g. service input, task/operation term, availability, etc.).

Acknowledgements

We are grateful to the myGrid team, in particular to Franck Tanoh and James Eales (the University of Manchester) for the evaluation of the results.

References

Al-Mubaid H. (2006). "Context-Based Technique for Biomedical Term Classification". *Proc.* of the 2006 IEEE Congress on Evolutionary Computation, CEC-2006, pp.5726-5733

Cannata N, E Merelli and RB Altman (2005). "Time to Organize the Bioinformatics Resourceome". PLoS Comput Biol 1(7): e76.

Collier N, C. Nobata and J. Tsujii (2000). "Extracting the Names of Genes and Gene Products with a Hidden Markov Model" Proc. of *COLING 2000*, pp. 201-207.

Frantzi K, S. Ananiadou and H. Mima (2000). "Automatic Recognition of Multi-Word Terms: The C-value/ NC-value method." International Journal on Digital Libraries 3(2): 115-130.

Grefenstette G (1994). "Exploration in Automatic Thesaurus Discovery". Springer, Vol. 278, 1994.

Hatzivassiloglou V, P.A.. Duboue and A. Rzhetsky (2001). "Disambiguating Proteins, Genes, and RNA in Text: A Machine Language Approach." Bioinformatics 17(1): 97-106.

Jensen JL, J. Saric and P. Bork (2006). "Literature mining for the biologist: from information retrieval to biological discovery". Nature Reviews Genetics

Kageura K and B. Umino (1996). "Methods of automatic term recognition: a review". Terminology 1996; 3:259–289.

Kittredge R (1982). "Sublanguages". Comput Linguist 8(2): 79-84.

Krauthammer M. and G. Nenadic (2004). "Term identification in the biomedical literature". Journal of Biomedical Informatics 37(6): 512-526.

Liu H. and C. Friedman (2003). "Mining Terminological Knowledge in Large Biomedical Corpora". Proc. of 8th PSB, p. 415-426

Nenadic G. and S. Ananiadou (2006). "Mining Semantically Related Terms from Biomedical Literature" ACM Transactions on ALIP 5(1): 22-43.

Oinn T, P. Li., DB Kell, C. Goble, A. Goderis, M. Greenwood, D. Hull, R. Stevens, D. Turi and J. Zhao (2007). "Taverna/myGrid: aligning a workflow system with the life sciences community." In Dennis et al. (Eds), Workflows for e-Science: scientific workflows for Grids, Springer, 300-319.

Sabou M, C. Wroe, C. Goble and G. Mishne (2005). "Learning Domain Ontologies for Web Service Descriptions: an Experiment in Bioinformatics." Proc of 14th Int. Conf. on WWW, p. 190-198

Spasic I. and S. Ananiadou (2004). "Using automatically learnt verb selectional preferences for classification of biomedical terms". Journal of Biomedical Informatics (Named Entity Recognition in Biomedicine), Vol. 37, No. 6, pp. 483-497

Spasic I., G. Nenadic and S. Ananiadou (2004). "Learning to Classify Biomedical Terms through Literature Mining and Genetic Algorithms". Proc. of *IDEAL 2004*, pp: 345-351.

Spasic I, S. Ananiadou and J. Tsujii (2005). "MaSTerClass: a case-based reasoning system for the classification of biomedical terms." Bioinformatics 21(11): 2748-2758.

Spasic I, D. Schober, SA Sansone, D Rebholz-Schuhmann, D Kell and N Paton (2008). "Facilitating the development of controlled vocabularies for metabolomics technologies with text mining." BMC Bioinformatics 9(Suppl 5):S5

Wolstencroft K, P. Alper, D. Hull, C. Wroe, P.W. Lord, R.D. Stevens and C. Goble (2007). "The myGrid Ontology: Bioinformatics Service Discovery". International Journal of Bioinformatics Research and Applications, 3(3):326 – 340, 2007.

Yan J (2002). "GeneSOM—self-organizing map package", Version 0.2-5, 2002. Available from http://cran.R-project.org.

Yeh A, A. Morgan, M. Colosimo and L. Hirschman (2005). "BioCreAtIvE Task 1A: gene mention finding evaluation". BMC Bioinformatics 2005; 6(Supp1):S2.

Lexical Properties of OBO Ontology Class Names and Synonyms

Elena Beisswanger **Michael Poprat** **Udo Hahn**

Jena University Language & Information Engineering (JULIE) Lab
Friedrich-Schiller-Universität Jena
D-07743 Jena, Germany
{elena.beisswanger|michael.poprat|udo.hahn}@uni-jena.de

Abstract

While the Open Biomedical Ontologies (OBO) are successfully used for manual database annotation purposes, their usefulness for automatic text mining remains to be shown. Crucial for OBO's suitability for natural language processing applications is the nature of the class names and synonyms provided and the way they are referred to or literally appear in the literature. Accordingly, our study investigates the lexical properties and the semantic ambiguity of these terms. In particular, we identify the number of OBO classes that can be recognized in two corpora by means of these terms, considering one corpus of full texts documents taken from PUBMED Central and one of MEDLINE abstracts. We found that 15% of all OBO classes could be identified in the MEDLINE corpus and 9% in the PUBMED Central corpus by case-insensitive string matching, including term variants and stemmed forms of terms. Interestingly enough, only nine out of 80 OBO ontologies account for 80% of the OBO classes that we were able to find.

1 Introduction

The Open Biomedical Ontologies (OBO) library[1] is a collection of publicly available biomedical ontologies hosted by the U.S. National Center for Biomedical Ontologies (NCBO). OBO ontologies cover different subdomains of biology and biomedicine, amongst others the anatomy of different model organisms, biomedical processes, molecular functions of gene products, sequence features, chemicals and experimental methods. The ontologies have been developed as controlled vocabularies for various data management tasks. For example, the Gene Ontology (GO),[2] the most prominent one of the OBO ontologies, was created for the functional annotation of genes and gene products. The aim of using shared controlled vocabularies such as the GO is to facilitate the interoperability of different but related biomedical databases across species.

OBO ontologies hold domain-specific knowledge in a structured way by using hierarchically organized classes and additional semantic relations. Classes come with a class name and are often supplemented with synonyms and textual definitions. While typically the class name is unambiguous and self-explaining, the synonyms are supplied to reflect the natural language use in documents and thus tend to be ambiguous. Besides the hierarchy defining *is-a* relation, many OBO ontologies provide complementary semantic relations (such as *part-of* and *develops-from*) to express complex domain-specific interdependencies (e.g., "cellular membrane" *part-of* "cell", "mature T cell" *develops-from* "immature T cell").

The OBO ontologies provide a huge amount of domain-specific vocabulary in terms of class names and their synonyms, which in their entirety we refer to as OBO *terms*. Although it is well known that natural language processing (NLP) may heavily benefit from access to biomedical domain knowledge (Spasic et al., 2005), re-use of OBO for NLP is rare. This might be due to the fact that the OBO terms are suspected to be rather artificial, utterly long and complex – taking the lexical properties of GO class names (McCray et al., 2002; Ogren et al., 2004) *pars pro toto* for OBO terms.

[1] http://obofoundry.org/

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

In order to find out whether this assumption is justified or not we here scrutinize on natural language properties of OBO terms. We do so by looking for matches of OBO terms in two natural language corpora, subsets of MEDLINE and PUBMED Central. We also incorporate another terminological umbrella system, the Unified Medical Language System (UMLS), that enjoys much greater acceptance in the biomedical NLP community than the OBO ontologies.

2 Related Work

Several studies focusing on the lexical nature of terms in domain-specific terminologies and ontologies have already been carried out. One of their main intentions is to find ways how these terms can be arranged in domain-specific lexicons that are easy to use by NLP engines.

Verspoor (2005) constructed a semantic lexicon based on terms that occur in both, the UMLS Metathesaurus[3] and the UMLS Specialist Lexicon.[4] Several matching strategies were applied to detect the UMLS terms in a domain-specific text corpus. Verspoor found a lexical overlap for 77% of the tokens in the corpus, though regarding the set of all different tokens (i.e., types) in the corpus only 3% were matched. The study did not explore whether the meaning of the terms found in a text really corresponded to the meaning of the targeted terms in the merged lexicon, i.e., the resolution of semantic ambiguity is left as an open issue.

McCray et al. (2001) also evaluated UMLS Metathesaurus terms regarding their usefulness for NLP. They identified different string properties that allow to predict how likely it is that a given term can be found in a domain-specific text corpus. In a follow-up study they explored the lexical properties of GO terms. In particular, they determined the number of GO terms that appear in the UMLS Metathesaurus as well, checked whether the terms fulfilled certain lexical properties that indicate their 'wellformedness' for NLP and looked for GO terms in a domain-specific text corpus (McCray et al., 2002). Another stream of work on GO terms investigates their compositionality (see, e.g., Ogren et al. (2004)) rather than their lexical features in relation to NLP tasks.

Our study resembles the one by McCray et al. (2002) but extends its scope of analysis from the GO to the whole of OBO. In addition, we go beyond the work of McCray et al. and Verspoor by investigating the semantic ambiguity of OBO ontology terms and collecting preliminary evidence for the usefulness of the OBO ontologies for a typical NLP task, namely coreference resolution.

3 Methods

We analyzed the OBO terms in several ways. First, we searched for OBO terms in a corpus of MEDLINE abstracts and PUBMED Central full-text documents using different string matching strategies. Second, we investigated the overlap of OBO terms with terms from the UMLS Metathesaurus and the UMLS Specialist Lexicon. Third, we checked the OBO terms for various lexical properties and compared the distribution of these properties over the terms to that over the terms found in the two corpora and the UMLS resources. Fourth, we investigated the potential for semantic ambiguity of OBO terms. Finally, we matched the OBO terms to the coreference annotations in a pre-release of the GENIA corpus kindly provided by the Tsujii Laboratory (U Tokyo).[5]

3.1 Corpus Construction

We downloaded the OBO ontologies in May 2008 from the OBO ftp site,[6] in OBO flat file format. 80 OBO resources were selected for investigation, excluding pure mapping or bridge files linking classes from one ontology to classes from another and files linking database entries to GO. In total, the OBO ontologies comprise 827,843 classes with different IDs. All class names and their synonyms were extracted from the 80 resources resulting in a set of 1,383,430 different OBO terms from which 791,699 are distinct class names. We included all kinds of synonyms in our study (exact, related, broad, and narrow ones) since we were interested in the complete inventory of terms provided by the OBO ontologies, rather than in exact denotations of ontology classes only.

We also downloaded the UMLS 2008AA release and used the UMLS METAMORPHOSYS tool[7] to create a customized Metathesaurus subset

[3] http://www.nlm.nih.gov/pubs/factsheets/umlsmeta.html

[4] http://www.nlm.nih.gov/pubs/factsheets/umlslex.html

[5] http://www-tsujii.is.s.u-tokyo.ac.jp/

[6] ftp://ftp.fruitfly.org/pub/obo

[7] http://www.nlm.nih.gov/pubs/factsheets/umlsmetamorph.html

OBO Ontologies	
number of ontologies	80
distinct classes	827,843
distinct class names	791,699
distinct class names and synonyms	1,383,430
UMLS Resources	
UMLS Metathesaurus terms	3,810,230
UMLS Specialist terms	624,955

Table 1: Term statistics for the OBO Ontologies and for the UMLS Metathesaurus and the UMLS Specialist Lexicon

which contained all UMLS source terminologies (in English). The UMLS Metathesaurus terms (a set of 3,810,230 concept names) were extracted from the 'Concept Names File' (MRCON). Furthermore, terms from the UMLS Specialist Lexicon were extracted from the 'Agreement and Inflection' file (LRAGR) resulting in a set of 624,955 different terms. Table 1 summarizes this data for the OBO ontologies and the UMLS resources.

The MEDLINE[8] download took place in February 2008 and included all records which contained an abstract and were entered between the years 2000 and 2008. 10% of these abstracts were randomly selected resulting in a corpus of 316,520 documents. In the following, we refer to this collection as the MEDLINE corpus.

PUBMED Central,[9] a full-text site for biomedicine and the life sciences, was downloaded in February 2008 and, again, 10% of the documents were randomly selected. This resulted in a corpus of 6,342 documents, henceforth the PMC corpus.

The (pre-release of the) GENIA coreference corpus is composed of 1,999 documents containing in total 46,067 annotations of coreferences. Table 2 presents an overview on the number of documents and tokens contained in the three corpora we used for our study. The token counts are provided for ease of comparison only and simply rely on counting whitespace-separated strings.

[8] http://www.ncbi.nlm.nih.gov/sites/entrez?db=pubmed

[9] http://www.pubmedcentral.nih.gov/

Corpus	Documents	Tokens, in total (different tokens)
MEDLINE	316,520	65,544,220 (1,656,388)
PMC	6,342	19,876,372 (631,601)
GENIA	1,999	460,334 (30,004)

Table 2: Number of documents and tokens in MEDLINE, PUBMED Central (PMC) and GENIA

3.2 Matching OBO with MEDLINE and PMC

In the first part of our study, we searched for OBO terms in the MEDLINE and the PMC corpus. All OBO terms were matched against the corpora using four different strategies, *viz.* exact match, case insensitive match, case insensitive match after adding simple term variants, and case insensitive match after adding term variants and stemmed terms.

The stemmed version of a term was created using the UEA stemmer[10] (for multi-token terms we only stemmed the last token of the term). The variants were generated using a combination of replacing multiple whitespaces, underscores and hyphens by blanks, and removing brackets and single quotes. In addition, we added variants for terms containing a comma followed by a space, such as *"liver arginase"* generated from *"arginase, liver"*, and created some ontology-specific variants (for GO molecular function terms, e.g., which contained the suffix *"activity"* a variant without that suffix was added to the set of OBO terms). The matching was carried out using the LINGPIPE EXACTDICTIONARYCHUNKER[11] and an integrated tokenizer.

We were interested in both, the number and nature of OBO terms that appear in the two text corpora, because these could shed light on the usage of domain-specific terms in natural language documents, and the number of OBO classes that could be identified based on these (slightly enriched) terms. Another interesting issue is whether the usage of domain-specific terminology differs in scientific abstracts (MEDLINE) from that in full-text documents (PMC). Therefore, we conducted the same matching study twice, on a corpus of abstracts and on a corpus of full-text documents.

[10] http://fizz.cmp.uea.ac.uk/Research/stemmer/

[11] http://alias-i.com/lingpipe/index.html

3.3 Matching OBO with UMLS

In the next part of the study we investigated the overlap between the OBO terms and the terms provided by the UMLS Metathesaurus and the UMLS Specialist Lexicon. All OBO terms were matched against the terms from the UMLS resources ignoring case sensitivity.

The OBO ontologies hold a huge amount of domain-specific vocabulary, but hardly contain any lexical information (syntactic, morphological, and orthographic information as contained in the UMLS Specialist Lexicon) or semantic typing (contained in the UMLS Metathesaurus in terms of UMLS Semantic Network type assignments). Therefore, the overlap of OBO and UMLS terminology gives evidence of how useful the UMLS could be as a source for lexical and semantic typing information to enrich the OBO ontologies. A substantial overlap of the OBO ontologies and the UMLS Metathesaurus could reasonably be expected since five important OBO ontologies are fully or at least partially covered in the UMLS Metathesaurus as well, namely the GO, the Foundational Model of Anatomy (FMA), the NCBI taxonomy, the NCI Thesaurus, and the Medical Subject Headings (MESH).

3.4 Analysis of Lexical Properties

Inspired by the study of McCray et al. (2001), we defined a set of lexical features to analyze the properties of the OBO terms. We chose the features *'holds at least one number'* (number), *'holds at least one parenthesis'* (parenthesis), *'holds at least one special character'* (special character) and *'average proportion of special characters'* (special characters (in %)) to estimate the character complexity of the terms. The features *'average number of characters'* (# characters) and *'average number of tokens'* (# tokens) were selected to quantify the length of terms. We also considered the features *'holds at least one underscore'* (underscore) and *'contains a comma followed by a blank'* (comma space). We identified the number of OBO terms revealing these features and compared it with the number of terms in the UMLS Metathesaurus, the UMLS Specialist Lexicon, and the number of OBO terms found in the MEDLINE and PMC corpus that exhibited these features (see Table 5).

The aim of this investigation was to gather evidence, first, which types of terms appear frequently in natural language documents, second, whether the nature of terms occurring in abstracts and in full-text documents differs markedly, and, third, whether the OBO terms reveal similar features as terms from the UMLS resources, or not.

3.5 Evaluating Semantic Ambiguity

Next, we analyzed the semantic ambiguity of OBO terms. We replaced all underscores in the OBO terms and turned them into lower case (case sensitivity and the use of underscores in class designators heavily depends on naming conventions) and selected exactly those terms that appeared in at least two different ontology classes and classified them as potentially ambiguous. Note that the existence of different identifiers for some ontology classes does not necessarily imply a semantic distinction (i.e., ambiguity) as well. Thus, the number of potentially ambiguous terms only constitutes an upper bound for the true number of ambiguous terms in the OBO ontologies.

The resulting list of terms was taken to determine the percentage of intra-ontology and cross-ontology ambiguity among all encountered ambiguities. We also identified the number of ambiguous terms that belonged to classes in a parent-child relationship. Our intention was to get further evidence whether the terms were really polysemous, or whether they belonged to two different ontology classes that, in fact, share the same meaning and should be merged, or whether they belonged to parent-child-related ontology classes due to a sloppy synonym assignment policy.

Semantic ambiguity in terms of polysemy (or homonymy) is a major problem when domain-specific terminology is taken into account by NLP applications (e.g., Liu et al. (2002) or Humphrey et al. (2006)). In particular, ambiguous terms decrease the performance of Named Entity Recognition (NER) tools which, in turn, affect the performance of all other NLP components relying on the output of the NER component. Thus, in order to assess the suitability of OBO terms for NLP, we consider the analysis of semantic ambiguity of the terms as an important problem.

3.6 Matching OBO with GENIA

In the last part of the study, we matched the OBO terms against the GENIA corpus, enriched with coreference annotations, and identified the number of exact and embedded matches of OBO terms

with coreference annotations (in case of an embedded match a term matches parts of a coreference annotation).

Coreferences are natural language expressions which share the same referent, i.e. refer to the same entity in the world, within or across sentences, though the denotations at the text surface are different (e.g., in *"IL-7 ... . This protein ..." "This protein"* corefers to *"IL-7"*). The process of determining proper coreference pairs is called coreference resolution and constitutes an important subtask in many NLP applications. Expressions that are potentially coreferent can be detected by syntactic analysis (within sentences) or by some sort of discourse memory (across sentences). Domain ontologies help restrict the number of candidates for resolution by providing both, categories for semantic typing of the expressions and semantic relations between these categories that can be exploited to infer the coreference of two expressions (Vlachos et al., 2006). The aim of our study was to get preliminary evidence for the usefulness of OBO classes for semantic typing of coreferent expressions.

4 Results

4.1 Results of the Corpus Matching Study

Applying case-insensitive term matching and incorporating term variants as well as stemming (henceforth, *IVS-matching*) we found a total number of 46.7M term matches in the MEDLINE corpus and 13.2M in the PMC corpus (cf. Table 2). The matches covered major proportions of the tokens in the two corpora (76% in the MEDLINE corpus and 70% in the PMC corpus, respectively), though only minor proportions of the set of all different tokens were covered (6% in the MEDLINE corpus and 9% in the PMC corpus, respectively).

Source	OBO Classes
OBO	827,843
MEDLINE Corpus	125,386 (15%)
PMC Corpus	76,718 (9%)
UMLS Metathesaurus	528,356 (64%)
UMLS Specialist Lexicon	128,704 (16%)

Table 3: Number of OBO classes associated with the OBO terms detected in the text corpora and the UMLS resources performing IVS-matching

The main focus of the corpus matching study was on the OBO classes that can be detected in the two corpora by means of the (slightly enriched) terms associated with them. We were able to identify (see Table 3) about 125,000 OBO classes in the MEDLINE corpus (15%) and almost 77,000 in the PMC corpus (9%).

For the MEDLINE corpus we carried out additional investigations. In order to find out which impact case normalization, variant generation and stemming had on the number of traceable OBO classes we conducted additional matching experiments (exact term matching, case insensitive matching, and case insensitive matching considering term variants). By applying exact term matching only the total number of matches dropped to 25.4M and the number of OBO classes that we were able to identify decreased by four percentage points to approximatelyly 94,000. This is only 75% of the classes found by IVS-matching. Case normalization and the generation of term variants accounted for major parts of the difference, while stemming had only little impact.

Next we investigated the importance of synonyms for detecting OBO classes in the corpus. We applied IVS-matching omitting all OBO synonyms. As a result, the number of term matches dropped to 26.8M and the number of traceable OBO classes was reduced by three percentage points to approximately 97,000. This incorporated only 78% of the classes found when considering synonyms.

Finally we analyzed from which OBO ontologies the OBO classes identified in the corpus came. The study revealed that a subset of only nine (out of 80) OBO ontologies accounted for more than three-fourths of the traceable OBO classes (namely the NCI Thesaurus, the NCBI Taxonomy, the MESH, the ontology for Chemical Entities of Biological Interest (CHEBI), the Gene Ontology (GO), the INOH Molecule Role ontology (IMR), the Human Developmental Anatomy ontology (EHDA), the Foundational Model of Anatomy (FMA), and the Disease Ontology, see Table 4). The same set of ontologies holds 76% of all OBO terms and 79% of all OBO classes.

4.2 Results of the UMLS Matching Study

The focus of the UMLS matching study was on the term overlap of the OBO ontologies with UMLS

OBO Ontologies	Identified Classes in MEDLINE Corpus
NCI Thesaurus	22.26%
NCBI Taxonomy	15.87%
MESH	10.25%
CHEBI	6.61%
GO	6.21%
IMR	5.61%
EHDA	4.54%
FMA	4.47%
Disease Ontology	4.22%
total	80.05%

Table 4: Nine out of eighty OBO ontologies account for more than 80% of the OBO classes identified in the MEDLINE corpus

resources. We found (see Table 3) approximately 763,000 OBO terms that also appeared in the UMLS Metathesaurus and about 119,000 that also appeared in the UMLS Specialist Lexicon, applying IVS-matching. The OBO terms matching Metathesaurus terms were associated with about 528,000 OBO classes (64%), those matching Specialist Lexicon terms with almost 129,000 (16%).

4.3 Results of the Analysis of Lexical Properties of Terms

We found (see Table 5) the OBO terms to be on the average three tokens and 27 characters long, about twice as long compared with the OBO terms detected in the MEDLINE and the PMC corpus, and also compared with the terms in the UMLS Specialist Lexicon. Furthermore, they contained more than twice as often numbers and special characters. Compared with the terms in the UMLS Metathesaurus OBO terms were shorter and contained less often non-alphabetic characters (numers, parentheses, special characters). In addition, we found that only a small proportion of OBO terms contained underscores and 17% of UMLS Metathesaurus terms were marked with the feature *'contains a comma followed by a blank'* (comma space). To locate these terms in the documents they had to be normalized first.

4.4 Results of the Evaluation of Semantic Ambiguity

We found (see Table 6) about 6% of the OBO terms to be associated with at least two ontology classes, which makes them potentially ambiguous. The average number of classes with which these ambiguous terms were associated was 2.7. Additional investigations revealed that about one third of the potentially ambiguous terms were ambiguous within one ontology and only a very small number of terms was associated with ontology classes that were related by an *is-a* relationship. When only class names were considered, i.e., synonyms were discarded, the proportion of ambiguous terms dropped by almost two percentage points and the average number of classes these terms were associated with turned out to be 2.6.

	Class Names	Class Names & Synonyms
OBO terms	1,040,119	2,013,354
ambiguous	44,193 (4.25%)	122,491 (6.08%)
intra-source ambiguous	3,816 (0.37%)	43,747 (2.17%)

Table 6: Number of ambiguous OBO class names and synonyms and intra-source ambiguity

4.5 Results of the GENIA Matching Study

When OBO terms were matched with the GENIA corpus by IVS-matching (see Table 7) about 317,000 matches were found. These included almost 5,000 exact matches of OBO terms with coreference annotations (some 10% of all 46,000 coreference annotations) and about 40,000 matches of OBO terms that were embedded in a coreference annotation.

5 Discussion and Conclusions

We presented a study of the lexical properties of terms contained in the OBO ontologies, identified an upper bound for the semantic ambiguity of these terms and investigated how useful they are to detect references to ontology classes in domain specific text corpora.

GENIA Corpus	
number of annotated corefs	46,067
total matches	317,493
exactly matched corefs	4,722
embedded matches	40,065

Table 7: Term matches found in the GENIA corpus

Feature	OBO terms (1,383,430)	OBO terms in MEDLINE (174,282)	OBO terms in PMC (82,786)	UMLS Metathesaurus terms (3,810,230)	UMLS Specialist terms (624,955)
number	32%	12%	11%	34%	1%
parenthesis	10%	1%	1%	21%	0%
comma space	3%	0%	0%	17%	0%
underscore	3%	0%	0%	0%	0%
special character	24%	11%	8%	47%	16%
special characters (in %)	10	5.70	5.11	12.69	4
# characters	27.12	14.43	12.38	34.83	14.59
# tokens	3	1.78	1.78	4.21	1.49

Table 5: Lexical properties of terms provided by the OBO ontologies, the UMLS Metathesaurus, the UMLS Specialist Lexicon, and those OBO terms identified in the MEDLINE and the PMC corpus via IVS-matching (percentages were rounded to integers)

As far as semantic ambiguity of OBO terms is concerned we were able to characterize upper bounds assessing how many OBO terms occurred in more than one OBO class. Future work will have to constrain the grade of ambiguity by mechanisms of ontology alignment revealing semantic equivalences between ontology classes with different identifiers. After having completed the alignment of ontologies the maximal number of pair-wise non-equivalent ontology classes to which a term is assigned will reflect exactly the number of different senses of the term (excluding senses beyond the scope of the OBO ontologies). At this stage the OBO ontologies would be a valuable sense inventory for biomedical terms that could be used as a basis for word sense disambiguation (WSD). WSD is another pending step of our work. Until now we simply looked for string-based matches of OBO terms in text corpora and the UMLS, but in case the terms had different senses we did not evaluate which sense of the term turned up in the text.

We examined whether there was a difference in the use of domain-specific terminology in scientific abstracts and in full-text documents. Much to our surprise, we identified less OBO classes in full-text documents (the PMC corpus) than in abstracts (the MEDLINE corpus). Furthermore, we found OBO terms matching full-text documents to be, on the average, shorter and less complex than those matching abstracts. An explanation could be that in full-text documents OBO classes are mentioned in terms of detailed descriptions, rather than by mentioning their name, whereas in abstracts knowledge is expressed in a much denser way, thus requiring more compact domain-specific terminology. However, the results for the PMC corpus are weaker than those for the MEDLINE corpus since it is three times smaller based on token counts. We are currently working on a second run of our experiments with a larger sized PMC corpus to reassess our results.

In the last part of our study we identified exact matches of OBO terms with GENIA coreference annotations, as well as embedded matches. What we did not do so far is checking whether in the case of two expressions matching OBO terms a semantic link between these expression could be inferred based on a semantic relation between OBO classes associated with the matched OBO terms. Only if such a relation could be found in the majority of cases coreference resolution systems would in fact benefit from the incorporation of OBO ontologies.

The main focus of our study was on the nature of OBO terms and their use in domain-specific corpora. Based on term matches we were able to infer which OBO classes were addressed in the corpora. We found that simple case normalization and term variant generation substantially increased the number of OBO classes that could be identified. Further, we observed that synonyms also played an important role.

However, the overall number of OBO classes that we were able to identify was extremely low (15% in the MEDLINE and 9% in the PMC cor-

pus). In fact, at the outset we expected a much higher proportion of OBO classes to be addressed in the domain-specific literature. Obviously, the textual realization of ontology terms is quite different from their appearance in ontologies. We assume this holds especially for those classes that are represented by rather lengthy terms containing many non-alphabetic characters. The problem of finding these additional mentions of ontology classes is well known for the Gene Ontology (cf. Blaschke et al. (2005)). Our data suggests that it also concerns the whole of OBO.

To be of real use for NLP applications we believe that the number of OBO classes that can currently be automatically detected in documents by string matching routines is still far to low. Our plans to enhance the number of traceable OBO classes are based on the following considerations. First, the OBO ontologies could be enriched with additional synonyms (presently only 39% of all OBO classes are provided with synonyms leaving room for improvement). Second, instead of simple string matching (exact or more liberal variants) more sophisticated mapping procedures should be developed particularly suited for detecting multi-token terms and terms that contain many non-alphabetic characters, such as names of chemical entities. Some specialized tools already exist that could be exploited for this purpose, such as METAMAP, a tool developed at the American National Library of Medicine to match UMLS terms and their textual forms (Aronson, 2001), or OSCAR3, a tool for the identification of chemical entities (Corbett and Copestake, 2008).

Acknowledgments

This work was partly funded by the German Ministry of Education and Research within the STEMNET project (01DS001A-C) and by the EC within the BOOTSTREP project (FP6-028099) under the 6th Framework Program.

References

Alan R. Aronson. 2001. Effective mapping of biomedical text to the UMLS Metathesaurus: The METAMAP program. In Suzanne Bakken, editor, *AMIA 2001 – Proceedings of the Annual Symposium of the American Medical Informatics Association*, pages 17–21. Washington, D.C., November 3-7, 2001. Philadelphia, PA: Hanley & Belfus.

Christian Blaschke, Eduardo Andres Leon, Martin Krallinger, and Alfonso Valencia. 2005. Evaluation of BIOCREATIVE assessment of task 2. *BMC Bioinformatics*, 6(Supplement 1: S16).

Peter Corbett and Ann Copestake. 2008. Cascaded classifiers for confidence-based chemical named entity recognition. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing (BioNLP2008)*, pages 54–62, Columbus, Ohio, USA, June.

Susanne M. Humphrey, Willie J. Rogers, Halil Kilicoglu, Dina Demner-Fushman, and Thomas C. Rindflesch. 2006. Word sense disambiguation by selecting the best semantic type based on Journal Descriptor Indexing: Preliminary experiment. *Journal of the American Society for Information Science and Technology*, 57(1):96–113.

Hongfang Liu, S. B. Johnson, and Carol Friedman. 2002. Automatic resolution of ambiguous terms based on machine learning and conceptual relations in the UMLS. *Journal of the American Medical Informatics Association*, 9(6):621–636.

Alexa T. McCray, Olivier Bodenreider, James D. Malley, and Allen C. Browne. 2001. Evaluating UMLS strings for natural language processing. In Suzanne Bakken, editor, *AMIA 2001 – Proceedings of the Annual Symposium of the American Medical Informatics Association*, pages 448–452. Washington, D.C., November 3-7, 2001. Philadelphia, PA: Hanley & Belfus.

Alexa T. McCray, Allen C. Browne, and Olivier Bodenreider. 2002. The lexical properties of the Gene Ontology (GO). In *AMIA 2002 – Proceedings of the Annual Symposium of the American Medical Informatics Association*, pages 504–508. San Antonio, TX, November 9-13, 2002. Philadelphia, PA: Hanley & Belfus.

Philip V. Ogren, Kevin B. Cohen, George K. Acquaah-Mensah, Jens Eberlein, and Lawrence Hunter. 2004. The compositional structure of GENE ONTOLOGY terms. In *PSB 2004 – Proceedings of the Pacific Symposium on Biocomputing 2004*, pages 214–225. Hawaii, USA, January 6-10, 2004. Singapore: World Scientific Publishing.

Irena Spasic, Sophia Ananiadou, John McNaught, and Anand Kumar. 2005. Text mining and ontologies in biomedicine: Making sense of raw text. *Briefings in Bioinformatics*, 6(3):239–251.

Karin Verspoor. 2005. Towards a semantic lexicon for biological language processing. *Comparative and Functional Genomics*, 6(1-2):61–66.

Andreas Vlachos, Caroline Gasperin, Ian Lewin, and Ted Briscow. 2006. Bootstrapping the recognition and anaphoric linking of named entities in *Drosophila* articles. In *PSB 2006 – Proceedings of the Pacific Symposium on Biocomputing 2006*, pages 100–111. Hawaii, USA, January 3-7, 2006. Singapore: World Scientific Publishing.

Testing Different ACE-Style Feature Sets for the Extraction of Gene Regulation Relations from MEDLINE Abstracts

Ekaterina Buyko **Elena Beisswanger** **Udo Hahn**

Jena University Language & Information Engineering (JULIE) Lab
Friedrich-Schiller-Universität Jena
D-07743 Jena, Germany
{ekaterina.buyko|elena.beisswanger|udo.hahn}@uni-jena.de

Abstract

Based on an introspective manual analysis how relations between biomedical entities surface literally in scientific abstracts, we investigate the performance of different feature sets for biomedical relation extraction in a supervised machine-learning setting. We start from fairly simple ACE-style ones and increasingly include domain-specific knowledge in these feature sets. This turns out to have beneficial effects on the extraction performance of the system under scrutiny.

1 Introduction

While systems for the recognition and interpretation of named entities have reached, by and large, a stable performance plateau at the 80% level, the extraction of relations between these entities lags far behind these figures. In the newswire domain, e.g., the *Automatic Content Extraction Program* (ACE) (Doddington et al., 2004) features the best system with 36.8% f-score in the detection of relation mentions.[1] This data is even underperfomed by the winning system of the *BioCreative 2 Protein Interaction Sub-Task* (IPS) (Hirschman et al., 2007), whose performance results settled at only 28.8% f-score. Although for both competitions strict real-world requirements were imposed on the task – the recognition and interpretation of all named entities involved, plus the recognition and interpretation of the associated relation (and, for the biomedical domain, the mapping of entities onto unique database identifiers) – relation extraction remains a challenging research problem under any conceivable conditions.

[1]Results are published at `http://www.nist.gov/speech/tests/ace/2007/`

Our approach to deal with the challenges arising from relation extraction (RE) in the biomedical domain is, first, to explore the possible reasons for the inherent hardness of this task through introspective manual text analysis. In Section 3 we discuss empirical phenomena underlying relation encodings in biomedical documents, including a large variety of patterns and reliance on inferential processes. We then consider the performance of a feature-based learning approach for RE. Since supervised machine learning relies on carefully crafted feature sets, we consider in Section 5 different varieties of these sets, starting from ones which have already proven useful in the newswire domain. We then explore, in a preliminary way though, the possible contribution of domain-specific features for further fine-tuning these feature sets. Encouraging as these results may be our current corpus (cf. Section 4) suffers from an unbalanced occurrence of (too few) positive examples (clearly an issue that has to be addressed in future work).

2 Related Work

As far as the state of the art in biomedical RE is concerned, the simplest method for the extraction of relations between named entities is the detection of bag-of-word-style *co-occurrences* of entities of interest within documents or sentences (e.g., Jenssen et al. (2001)). Co-occurrence-based approaches are characterized by a high recall at the cost of an extremely low precision. Furthermore, the type and direction of relation usually cannot be determined. RE approaches that focus on higher precision but often suffer from weaker recall are based on *manually defined patterns* (e.g., Blaschke et al. (1999)). Some pattern-based

approaches exploit morpho-syntactic and syntactic information and are based on *automatically learning RE patterns* from large corpora (e.g, Hakenberg et al. (2005), Huang et al. (2004)). These methods provide higher recall than those based on manually defined patterns. *Rule-based RE approaches* typically exploit full parse data of sentences and additional semantic information (e.g., Yakushiji et al. (2001), Saric et al. (2004), Fundel et al. (2007)).

In the newswire domain, supervised approaches currently dominate RE. This is partly due to the availability of large annotated corpora such as ACE (Doddington et al., 2004) that can be used for the training of machine learning models, using, e.g., Support Vector Machines (SVM) or Maximum Entropy (ME) models. The systems either exploit SVM kernels especially designed for the comparison of syntactic trees (e.g., Zelenko et al. (2003), Bunescu and Mooney (2005)) or they incorporate a variety of lexical, morpho-syntactic and syntactic features (e.g., Kambhatla (2004), Zhou et al. (2005)).

Considering RE in the biomedical domain, to the best of our knowledge, there are few studies which deal exclusively with gene regulation. Yang et al. (2008) focus on the detection of sentences that contain mentions of transcription factors (proteins regulating gene expression). They aim at the detection of new transcription factors, while relations are not taken into account. In contrast, Saric et al. (2004) extract gene regulatory networks and achieve in the RE task an accuracy of up to 90%. They disregard, however, ambiguous instances that potentially lower recall (no recall measures are reported in this work). The *Genic Interaction Extraction Challenge* (Nédellec, 2005) was organized to determine the state-of-the-art performance of systems designed for the detection of gene regulation interactions. The best system achieved a performance of about 50% f-score.

The LLL corpus which was especially created for the *Genic Interaction Extraction Challenge*, the AIMED[2] and the BIOINFER corpus (Pyysalo et al., 2007) are considered as the standard corpora in the biomedical domain for evaluation of RE performance (Pyysalo et al., 2008). They are usually used for the development and evaluation of feature-based systems (e.g., Katrenko and Adriaans (2006), Sætre et al. (2007)). Katrenko and Adriaans (2006) report on experiments in which they reached f-score peaks on the AIMED corpus with 54.3% and on the LLL corpus with 72.4%, respectively. It should be noted that the experimental settings of the evaluations are not always clear. Therefore, Airola et al. (2008) suggested to indicate the evaluation settings of the instance-, sentence- or document-wise evaluation and proposed to use the latter as a default evaluation setting.

In our work we decided to use supervised machine learning models for gene regulation data and investigate whether the experience gained in the newswire domain can be applied to the biomedical domain as well. We chose the system presented by Zhou et al. (2005) that achieved competitive results of 74.7% in RE on ACE.

3 Textual Patterns for Gene Regulation

Very briefly, the regulation of gene expression can be described as the process that modulates the frequency, rate or extent of gene expression, where gene expression is the process in which the coding sequence of a gene is converted into a mature gene product or products, namely proteins or RNA (taken from the definition of the Gene Ontology class *Regulation of Gene Expression*, GO:0010468).[3] Transcription factors, cofactors and regulators are proteins that play a central role in the regulation of gene expression.

To get acquainted with the textual appearance of gene regulation relations, we manually analyzed 50 randomly extracted sentences from MEDLINE abstracts (see Section 4 for a detailed description of that collection) in order to find recurrent patterns by which gene regulation relations are literally expressed in sublanguage documents. We found 58 encoded relations and discovered in this set at least thirteen stable patterns how gene regulation relations surface in texts. In the following list we rank these patterns by their frequency in the set (in descending order).

1. [`Agent`] V-active [`Patient` Action-NN]
   "IclR also represses the expression of iclR"
2. [`Patient` Action-NN] V-passive [`Agent`]
   "yeiL expression is positively activated by Lrp"

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

[3] `http://www.geneontology.org/`

3. [Agent] - Action-JJ [Patient]
   "SlyA-induced proteins"
4. [Agent] *is an* Actor-NN *for* [Patient]
   "IclR is a repressor for the Escherichia coli aceBAK operon"
5. [Agent Action-NN] V-active [Patient Action-NN]
   "Elevation of ppGpp levels in growing cells ... triggered the induction of all usp genes."
6. [Agent] V-active (*bind to*) [Patient] *promoter / site*
   "ZntR is a trans-acting repressor protein that binds to the znt promoter region"
7. [Agent] *is required / essential / involved in* [Patient Action-NN]
   "rpoS function is essential for bgl silencing"
8. [Action-NN *of* Patient] *by* [Agent]
   "transcription repression of the Escherichia coli acetate operon by IclR"
9. [Patient Action-NN] V-active [Agent]
   "Expression of the tau and ssu genes requires the LysR-type transcriptional regulatory proteins CysB and Cbl"
10. *Promoter of* [Patient] *contains binding site for* [Agent]
    "The promoters of the mar/sox/rob regulon of Escherichia coli contain a binding site (marbox) for the homologous transcriptional activators MarA, SoxS and Rob."
11. [Patient Action-NN] V-passive (*caused by*) [Agent]
    "bgl silencing caused by C-terminally truncated H-NS"
12. [Agent Action-NN] V-active (*cause*) [Patient Action-NN]
    "Disruption of cueR caused loss of copA expression"
13. [Action-NN Patient] *is under control of* [Agent]
    "Synthesis of Cbl itself is under control of the CysB protein"

The two most frequent patterns that contain the mention of regulation verbs (V-active, V-passive) cover a large amount of relation instances (1. (15 relations), 2. (11 relations)). The use of adjectives and Actor nominalizations (Actor-NN such as *'regulator'*) are other frequent patterns in the expression of gene regulation relations (3. (6 relations), 4. (4 relations)). Uncertain expressions such as *'be essential'*, *'be involved in'* are used for the description of unspecified gene regulation relations (7. (3 relations)). Besides regulation verbs and their nominalizations and adjectives indicating requirements or dependencies, authors frequently describe the molecular process of the binding of a transcrition factor to a gene region (promoter) or provide information that a gene contains a binding site for a transcription factor (6. (3 relations), 10. (1 relation)). Another pattern group contains causal relations between molecular processes in which gene and transcription factors are involved (5. (3 relations), 11. (1 relation), 12. (1 relation)). Five out of 58 relations could not be classified into the featured thirteen patterns.

We see two challenges for the correct detection of gene regulation relations in sentences (given adequate syntactic structures are available): (1) the detection of *is-a* relations between mentions of entities participating in regulation process, and (2) inferential processes on biomedical knowledge for correctly distinguishing between positive and negative regulation processes.

The first issue concerns the frequent occurrence of appositions and predicate noun relations between participants, and the use of anaphoric mentions of participants in the regulation process. For example, in *"zntR gene encodes a putative regulatory protein that controls the expression of the znt operon"* we have an *is-a* relation between *'zntR gene'* and *'putative regulatory protein'*.

The second challenging issue concerns the correct detection of the category of gene regulation relations (*positive*, *negative* or *unspecified*). Experimental environments for gene regulation detection often involve genetic modifications of transcription factors. By means of these genetic modifications and the expression levels of other genes, researchers implicitly draw conclusions about the role of the transcription factor in the gene regulation processes. The sentence *"The production of C51 microcin decreased or was absent in rpoS, crp and cya mutant cells."* contains a description of the decrease of *'C51'* expression level. The fact that *'rpoS'*, *'crp'* and *'cya'* genes are inactivated (*'mutant cells'*) leads to the conclusion that they positively regulate *'C51'*. The detection of the correct type of relation requires knowledge about experimental conditions (indicated here by *'mutant cells'*) and inferences on biomedical knowledge.

4 Corpus Annotation

In this section we introduce the *JULIE Lab Gene-Reg corpus* which consists of MEDLINE abstracts dealing with gene regulation in *E. Coli*. It provides three types of semantic annotations:

- named entities involved in gene regulatory processes, such as TFs (transcription factors, cofactors and regulators) and genes,
- pairwise relations between TFs and genes,
- triggers (e.g., clue verbs) essential for the description of gene regulation relations

For all three annotation levels the annotation vocabulary was taken from the *Gene Regulation Ontology* (GRO) (Beisswanger et al., 2008). GRO describes gene regulation processes occurring on the intra-cellular level (such as the binding of transcription factors to DNA binding sites) and physical entities that are involved in these processes (such as genes and transcription factors).

A set of 32,155 abstracts was compiled from MEDLINE based on a query including the MESH terms *Escherichia coli*, *Gene Expression* and *Transcription Factors* (amongst others). From this set we randomly selected a corpus of 314 abstracts for manual annotation.

4.1 Named Entities

All abstracts in the corpus were annotated manually by a graduate student of biology considering the semantic categories enumerated in Table 1 which also gives the annotation counts (for definitions of the selected categories see GRO).[4]

Named Entity Category	Annotations
Transcription Factor	2496
Transcription Cofactor	14
Transcription Regulator	40
Gene	2547
Gene Group	1180
Gene (anaphoric)	24
Gene Group (anaphoric)	71

Table 1: Number of entity annotations per semantic category

To assess the Inter-Annotator Agreement (IAA) for the entity annotation a second graduate student of biology annotated a subset of 248 abstracts. For this subset the IAA was computed applying three standard IAA measures for the NE task: Strict IAA (69% (R), 62% (P), 65% (F)), Correct-Span IAA (74% (R), 76% (P), 72% (F)) and Correct-Category IAA (79% (R), 81% (P), 80% (F)).

Additionally, the biologist annotated anaphoric mentions of *Gene* and *Gene Group* entities (see also Table 1). The motivation of this annotation task was to provide more instances of entity mentions for the annotation of gene regulation relations. Anaphoric mentions were only annotated in sentences containing gene regulation relations.

4.2 Relations

The corpus of MEDLINE abstracts as described in Section 4 was also annotated with relations by a graduate biologist in a two-step annotation process. In a first step, trigger words indicating mentions of gene regulation processes were annotated. In a second step, the relations between genes and TFs (affecting the expression of the gene) were annotated. Next, we describe the two-step annotation process in more detail.

4.2.1 Annotation of Trigger Words

In preparation of the trigger word annotation, one biologist and one linguist manually screened the abstracts in the corpus and compiled a list of verbalizations of molecular processes that frequently occurred in the description of gene regulation relations. These processes were grouped in five categories enumerated in Table 2 based on conceptualizations and definitions from the GRO.

Trigger words indicating textual mentions of the listed processes were annotated with the corresponding categories. A trigger is any literal verbal form that clearly signals the occurrence of a particular molecular process. Trigger words are basically main verbs, verb nominalizations and adjectives. For example, the sentence *"H-NS and*

Semantic Category	Annotations
GeneExpression	495 (15)
TranscriptionOfGene	46 (12)
RegulationOfGeneExpression (unspecified)	896 (82)
PositiveRegulationOfGeneExpression	835 (110)
NegativeRegulationOfGeneExpression	441 (93)

Table 2: Number of trigger word annotations per semantic category (unique annotations are in brackets)

[4] http://www.ebi.ac.uk/Rebholz-srv/GRO/GRO.html

StpA proteins stimulate expression of the maltose regulon in Escherichia coli." contains two trigger words: first, *'stimulate'* is a trigger for a process in the category *PositiveRegulationOfGeneExpression*, second, *'expression'* is a trigger for a process belonging to the category *GeneExpression*.

4.3 Annotation of Gene Regulation Relations

In the second step, the graduate biologist annotated pairwise relations between genes and transcription factors, cofactors and general regulators affecting the expression of the gene. This annotation was based on the GRO class *RegulationOfGeneExpression* with its two sub-classes *PositiveRegulationOfGeneExpression* and *NegativeRegulationOfGeneExpression*. The concept *RegulationOfGeneExpression (unspecified)* was used for the annotation of gene regulation processes that could not be specified as either a positive or a negative regulation. We chose single sentences as annotation context for this task so that only those textual mentions of gene regulation relations and their participants were annotated which occurred within the same sentence.

A relation instance contains two arguments, *Arg1* and *Arg2*. *Arg1* is occupied by the *agent*, i.e., the entity that plays the role of modifying the gene expression. *Arg2* is occupied by the *patient*, i.e., the entity of which the expression is modified. While agents are proteins that regulate the expression of genes, patients are typically the genes of which the expression is regulated by the agent. The sentence *"The uxuAB operon is negatively controlled by the uxuR and exuR regulatory gene products."* denotes a *NegativeRegulationOfGeneExpression* relation between the gene *'uxuAB'* and two transcription factors, viz. *'uxuR'* and *'exuR'*.

A set of 65 randomly selected abstracts was annotated by the second graduate student of biology for determing the IAA. A Strict IAA of 82% (R), 84% (P), 83% (F) was achieved for the task of the trigger annotation. An IAA of 78.4% (R), 77.3% (P), 77.8% (F) was measured for the task of correct identification of the pair of interacting named entities in gene regulation processes, while 67% (R), 67.9% (P), 67.4% (F) were achieved for the identification of interacting pairs plus the 3-way classification of the interaction relation.

Semantic Category	Annotations
RegulationOfGeneExpression (unspecified)	408
PositiveRegulationOfGeneExpression	455
NegativeRegulationOfGeneExpression	272

Table 3: Number of gene regulation relation annotations per semantic category

5 Methods

The patterns discussed in Section 3 already reveal the diversity how gene regulation relations surface literally in texts. For the automatic extraction of gene regulation relations, we pursue a feature-based approach to RE that incorporates diverse lexical, syntactic and semantic features. We, first, selected features that had already proved useful in the detection of relationships between entities in the newswire domain and were evaluated on the ACE RE corpus (Doddington et al., 2004). These features were intensively explored in the work of Zhou et al. (2005). As a classification model we chose the Maximum Entropy model implementation in MALLET.[5]

5.1 Features for Relation Extraction

Zhou et al. (2005) investigated eight classes of features suited for RE: words, entity type, mention level, overlap, base phrase chunking, dependency tree, parse tree and semantic resources. We chose seven classes of features (excluding semantic resources that were suited only for the newswire domain). In the following, we will briefly introduce these features. (for more detailed information, see Zhou et al. (2005)). As a substitute for Zhou et al.'s semantic resources, we incorporated a semantic feature class that exploits information about trigger word occurrence in the sentence (in the full parse tree path).

In the following we distinguish between two entity mentions in pairwise relations, i.e., E1 and E2. E1 is the entity mention that occurs first in the sentence (before E2). If one of the mentions includes another entity mention, then the entity mention with a larger span is classified as E1.

5.1.1 Words Features

This feature class covers four categories of words: (1) the words of both entity mentions, (2)

[5] MALLET is available at `http://mallet.cs.umass.edu/index.php/Main_Page`

the words between the entity mentions, (3) the words before E1, (4) the words after E2. For both mentions, head words and their combination are considered. The window for the words before E1 and after E2 has a size of two words.

5.1.2 Entity Features

Entity features account for combinations of entity types, flags indicating whether mentions have an overlap, and their mention level. For the latter, we distinguish between name and anaphoric mentions.

5.1.3 Base Phrase Chunking Features

The chunking features are concerned with the head words of the phrases between the two entity mentions. Zhou et al. (2005) show that shallow parsing features play a critical role for RE. This feature class covers four categories of phrases: (1) the phrases of both entity mentions, (2) the phrases between the entity mentions, (3) the phrases before E1, (4) the phrases after E2.

5.1.4 Full Parsing Features

This class of features deals with full parse tree information. We included in our work five from eight features presented by Zhou et al. (2005) excluding dependency features that concern dependencies derived from full parse tree as they were unreproducible from the paper's descriptions. We selected, however, features that exploit constituency-based parsing and indicate whether mentions are in the same noun, prepositional or verbal phrase. The path of phrase labels (without duplicates) between two entity mentions and the path enriched with phrase head information are considered as well.

5.1.5 Relational Trigger Words and Keywords

This newly added feature class accounts for the connection of trigger words and mentions in a full parse tree. We exploit features indicating whether the top phrase in the parse path between the entity mentions contains a *RegulationOfGeneExpression* trigger or one of its sub-type triggers, and whether *TranscriptionOfGene/GeneExpression* triggers occur in the same noun phrase as entity mentions. To acount for the influence of the experimental intervention context on the proper detection of the gene regulation relation, we checked whether keywords describing experimental interventions (e.g., *'mutant'*, *'deletion'*) (altogether, 56 keywords) co-occur in the same noun phrase with E1 or E2. For the evaluation of these features on the AIMED corpus we compiled a dictionary of interaction event triggers from our regulation trigger list and terms used by Fundel et al. (2007).

6 Experiments and Results

For the evaluation of our feature-based approach to gene regulation RE, we performed a ten-fold sentence-wise cross-validation on the GENEREG and the AIMED corpus. For the evaluation of the RE task we used original annotations of named entities and relational trigger words (the AIMED corpus was automatically tagged for event trigger words (cf. Section 5.1.5)). Anaphoric mentions of entities were included in the evaluation as well. We considered thus only the detection of a gene regulation relation between two entity mentions. As the regulation relation is an asymmetric one, we distinguish between *ARG1-relation-ARG2* and *ARG2-relation-ARG1*, i.e., the order in which the entities appear in the sentence. The overall evaluation results reflect the mean of both relations. The ten-fold cross-validation was done for (1) the binary classification of *RegulationOfGeneExpression* relation and (2) the 3-way classification *PositiveRegulationOfGeneExpression*, *NegativeRegulationOfGeneExpression*, and *RegulationOfGeneExpression (unspecified)*.[6]

The GENEREG corpus was enriched with morpho-syntactic and syntactic information. For POS tagging, chunking and parsing we used the re-trained OPENNLP tool suite.[7] These tools had previously been re-trained (Buyko et al., 2006) on the GENIA corpus (Ohta et al., 2002).

We evaluated on two feature sets: *Feature Set 1* (Zhou et al., 2005) and *Feature Set 2* that contains, in addition, the features exploiting relational triggers (see 5.1.5). The evaluation results clearly indicate that the straightforward porting of RE feature types from the newswire domain to the specialized biomedical domain does not provide fully satisfactory results (see Table 4). The addition of the domain-specific features (relational trigger words) increases the performance by 3 percentage points for the detection of the generic gene regulation relation (63.0%), and by nearly 14 percentage points for the detection of the specific

[6] *RegulationOfGeneExpression* does not contain specifications as to whether it is positive or negative.

[7] `http://opennlp.sourceforge.net/`

Semantic Category	Feature Set 1			Feature Set 2		
	R	P	F	R	P	F
RegulationOfGeneExp. (generic)	50.0	76.2	60.0	55.0 (78.6)	75.4 (56.8)	63.0 **(65.5)**
RegulationOfGeneExp. (pos.)	28.9	60.6	37.2	39.1 (56.2)	66.0 (48.8)	47.3 (**50.0**)
RegulationOfGeneExp. (neg.)	18.0	51.8	24.4	29.5 (41.2)	60.5 (51.3)	37.6 (**44.5**)
RegulationOfGeneExp. (unspec.)	10.8	31.7	14.9	28.5 (45.6)	60.9 (47.3)	37.3 (**44.6**)
Overall (pos./neg./unspec. at once)	19.5	55.6	28.2	31.5 (46.6)	65.0 (49.3)	42.0 **(47.3)**
AIMED	42.5	66.8	**51.5**	42.9	64.8	51.3

Table 4: Results of Gene Regulation Relation Extraction on the GENEREG (lines 3–7) and AIMED (line 8) corpus (reduction of negative examples (under-sampling) in brackets)

gene regulation relations (42.0%). Surprisingly, the evaluation on the AIMED corpus reveals that the incorporation of such semantic features does not enhance the over-all performance.

The error analysis with respect to specific relations revealed that the relation representations covered by the most frequent patterns (see Section 3) are correctly detected. Still, the main trouble here is the incorrect analysis of coordinated structures by the OPENNLP parser. In the incorrectly parsed sentence *"NarL Expression from the Escherichia coli nrf operon promoter is activated by the anaerobically triggered transcription factor, FNR, and by the nitrate/nitrite ion-controlled response regulators, NarL or NarP, but is repressed by the IHF and Fis proteins."* only one out of the five relation mentions was detected (the relation between *'nrf'* and *'FNR'*).

Another prominent failure source is in the frequent occurrence of anaphoric expressions within sentences in abstracts. The sentence *"The expression of the appY gene is induced immediately by anaerobiosis, and this anaerobic induction is independent of Fnr, and AppY, but dependent on ArcA"* contains an anaphoric expression *"this anaerobic induction"* that is crucial for the detection of the relation between transcription factors and the induced gene.

Furthermore, the rather weak results for the classification of specific gene regulation relations are partly due to inferences required for the proper detection of the category of the gene regulation relations. Currently, we handle these inferences (inappropriately though) only by exploiting keywords in the features. In order to more adequately deal with this problem we will focus, in future work, on a hybrid approach which includes at least a modest level of inferential capabilities.

One of the problems we see in the training data is the severe imbalance of positive and negative instances. The corpus contains about 9,000 negative instances and 1,135 positive instances only. Such an imbalance may cause serious learning problems, and is reflected already by the low performance (e.g, the particularly low recall of the minority class). If the classifier is uncertain, it typically predicts the majority class (in our case, the negative class label).

The sparseness of positive learning examples can be reduced by balancing the number of positive and negative examples in the training data. Two sampling schemes are usually applied here, over-sampling and under-sampling. The aim of over-sampling is to increase the number of the minority class instances, the goal of under-sampling is to reduce the number of the majority class instances. In our experiments, we chose under-sampling by randomly reducing negative examples in the training data down to the double number of the positive instances. The evaluation results for under-sampling reveal a substantial gain in the f-score up to 5 percentage points (see Table 4). In these runs we achieved the best performance of 65.5% for the detection of the generic gene regulation relation, and 47.3% for the extraction of specific gene regulation relations.

7 Conclusions and Future Work

We presented here a supervised approach to relation extraction in the biomedical sub-domain of gene regulation. The main contributions of this paper are in the descriptive analysis of the inherent hardness of this task and in tests of different feature sets for the extraction of gene regulation relations. Our evaluation results reveal that the straightforward porting of feature types that have

already proven useful in the newswire RE should be fine-tuned by integrating domain-specific feature sets. The balancing of the training data and a comparison of our approach with state-of-the art rule-based systems (e.g., Fundel et al. (2007)) are the focus of our future work.

Acknowledgments

This work was partly funded by the EC within the BOOTSTREP project (FP6-028099) under the 6th Framework Program.

References

A. Airola, S. Pyysalo, J. Björne, T. Pahikkala, F. Ginter, and T. Salakoski. 2008. A graph kernel for protein-protein interaction extraction. In *Proceedings of the BioNLP Workshop at ACL 2008*, pages 1–9. Columbus, Ohio, USA, June 19, 2008.

E. Beisswanger, V. Lee, J. Kim, D. Rebholz-Schuhmann, A. Splendiani, O. Dameron, S. Schulz, and U. Hahn. 2008. Gene Regulation Ontology (GRO): Design principles and use cases. In *Proceedings of the MIE 2008 Conference*, pages 9–14. Gøteborg, Sweden, 25-28 May 2008.

C. Blaschke, M. Andrade, C. Ouzounis, and A. Valencia. 1999. Automatic extraction of biological information from scientific text: Protein-protein interactions. In *Proceedings of the ISMB'99*, pages 60–67. Heidelberg, Germany, August 6-10, 1999.

R. Bunescu and R. Mooney. 2005. A shortest path dependency kernel for relation extraction. In *Proceedings of the HLT-EMNLP'05*, pages 724–731. Vancouver, B.C., Canada, October 6-8, 2005.

E. Buyko, J. Wermter, M. Poprat, and U. Hahn. 2006. Automatically adapting an NLP core engine to the biology domain. *Proceedings of the Joint Bio-Ontologies and BioLINK Meeting at ISMB 2006*, pages 65–68. Fortaleza, Brazil, August 5, 2006.

G. Doddington, A. Mitchell, M. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. 2004. The Automatic Content Extraction (ACE) Program: Tasks, data, & evaluation. In *Proc. LREC 2004*, pages 837–840. Lisbon, Portugal, 26-28 May 2004.

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

J. Hakenberg, U. Leser, C. Plake, H. Kirsch, and D. Rebholz-Schuhmann. 2005. LLL'05 Challenge: Genic Interaction Extraction – Identification of language patterns based on alignment and finite state automata. *Proc. 4th Learning Language in Logic Workshop*, pages 38–45. Bonn, August 2005.

L. Hirschman, M. Krallinger, and A. Valencia, editors. 2007. *Proceedings of the Second BioCreative Challenge Evaluation Workshop*. Madrid: CNIO Centro Nacional de Investigaciones Oncológicas.

M. Huang, X. Zhu, D. Payan, K. Qu, and M. Li. 2004. Discovering patterns to extract protein-protein interactions from full texts. *Bioinformatics*, 20(18):3604–3612.

T. Jenssen, A. Lægreid, J. Komorowski, and E. Hovig. 2001. A literature network of human genes for high-throughput analysis of gene expression. *Nature Genetics*, 28:21–28.

N. Kambhatla. 2004. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In *Proc. ACL 2004 on Interactive Poster and Demonstration Sessions*, pages 22–25. Barcelona, Spain, July 21-26, 2004.

S. Katrenko and P. Adriaans. 2006. Learning relations from biomedical corpora using dependency trees. *Proceedings of the KDECB 2006 Workshop*, pages 61–80. Ghent, Belgium, May 10, 2006.

C. Nédellec. 2005. Learning Language in Logic: Genic interaction extraction challenge. *Proceedings of the 4th Learning Language in Logic Workshop*, pages 31–37. Bonn, Germany, August 2005.

T. Ohta, Y. Tateisi, and J.-D. Kim. 2002. The GENIA corpus: An annotated research abstract corpus in molecular biology domain. In *Proceedings of the HLT 2002*, pages 82–86. San Diego, CA, USA, March 24-27, 2002.

S. Pyysalo, F. Ginter, J. Heimonen, J. Björne, J. Boberg, J. Järvinen, and T. Salakoski. 2007. BIOINFER: A corpus for information extraction in the biomedical domain. *BMC Bioinformatics*, 8(50).

S. Pyysalo, A. Airola, J. Heimonen, J. Björne, F. Ginter, and T. Salakoski. 2008. Comparative analysis of five protein-protein interaction corpora. *BMC Bioinformatics*, 9(Supplement 3):S6.

R. Sætre, K. Sagae, and J. Tsujii. 2007. Syntactic features for protein-protein interaction extraction. *Short Paper Proceedings of the LBM 2007*, Singapore, December 6-7, 2007.

J. Saric, L. Jensen, P. Bork, R. Ouzounova, and I. Rojas. 2004. Extracting regulatory gene expression networks from PUBMED. In *Proceedings of the ACL 2004*, pages 191–198. Barcelona, Spain, July 21-26, 2004.

A. Yakushiji, Y. Tateisi, Y. Miyao, and J. Tsujii. 2001. Event extraction from biomedical papers using a full parser. *Proceedings of the PSB 2001*, pages 408–419. Maui, Hawaii, USA. January 3-7, 2001.

H. Yang, G. Nenadic, and J. Keane. 2008. Identification of transcription factor contexts in literature using machine learning approaches. *BMC Bioinformatics*, 9(Supplement 3):S11.

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

G. Zhou, J. Su, J. Zhang, and M. Zhang. 2005. Exploring various knowledge in relation extraction. In *Proceedings of the ACL'05*, pages 427–434. Ann Arbor, MI, USA, 25-30 June 2005.

Classifying Disease Outbreak Reports Using N-grams and Semantic Features

Mike Conway, Son Doan, Ai Kawazoe and Nigel Collier
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan
{mike,doan,zoeai,collier}@nii.ac.jp

Abstract

This paper explores the benefits of using n-grams and semantic features for the classification of disease outbreak reports, in the context of a text mining system — BioCaster — that identifies and tracks emerging infectious disease outbreaks from online news. We show that a combination of bag-of-words features, n-grams and semantic features, in conjunction with feature selection, improves classification accuracy at a statistically significant level when compared to previous work. A novel feature of the work reported in this paper is the use of a semantic tagger — the USAS tagger — to generate features.

1 Introduction

Reliable document classification is an important pre-processing stage in many Information Extraction and Text Mining systems (Feldman and Sanger, 2007).[1] This paper compares the performance of a document representation based on highly discriminating unigrams, bigrams, trigrams and semantic features, against a representation based on unigram and Named Entity (NE) features used by Doan et al. (2007), for the classification of disease outbreak reports. While the document representation used by Doan et al. (2007) performed well for this task, a statistically significant improvement in performance was achieved using a representation based around n-grams and semantic features. A novel feature of this work is the use of a general purpose semantic tagger to generate features.

Following a discussion of related work in section 2, we describe in section 3 the feature sets used in this work and how they were derived. Section 4 sets out the methodology used, while section 5 presents results, and some discussion of those results. The final section outlines some broad conclusions and areas for future work.

2 Background

The BioCaster Corpus is a product of a wider project designed to aid in the surveillance and tracking of infectious disease outbreaks using text mining technology. The BioCaster system (Doan et al., 2008) scans online news reports for stories concerning infectious disease outbreaks. An article is of interest if it contains information about newly emerging infectious diseases of potential international significance, such as, the spread of diseases across international borders, the deliberate release of a pathogen, and so on. There are two methods that users can exploit to explore extracted data. First, the pre-interpreted information is publicly available on a web portal (built on Google Maps).[2] Second, registered users can opt to receive information (via email) on diseases, countries or other alerting conditions that interest them. According to Heymann et al. (2001), around 65% of disease outbreaks are first identified from the web.

The BioCaster gold standard corpus is a collection of 1000 news articles selected from the WWW, and subsequently manually categorized and annotated by two PhD students at the National Institute of Informatics (see Figure 1 for

[1] Cohen and Hersh (2005) includes a brief review of important work on text classification in the biomedical domain.

[2] The publicly accessible face of the BioCaster system is a visualization tool called *Global Health Monitor*. It is accessible at the BioCaster Portal (http://www.biocaster.nii.ac.jp).

```
<DOC id="000101" language="en-us"
source="WHO" domain="health"
subdomain="disease" date=2007/3/2
relevancy="publish">
<NAME cl="DISEASE">Avian
Flu</NAME> situation in <NAME
cl="LOCATION">Vietnam</NAME> update
21
 <NAME cl="TIME">16
June 2005</NAME><NAME
cl="ORGANIZATION">WHO</NAME>
is aware of media reports
that <NAME cl="PERSON"
case="true" number="many">six
additional patients</NAME><NAME
cl="CONDITION">infected</NAME>
with <NAME cl="DISEASE">H5N1
avian influenza</NAME> are
undergoing treatment in a <NAME
cl="LOCATION">Hanoi</NAME>
hospital and that <NAME cl="PERSON"
case="true" number="one">a health
care worker</NAME> at the same
hospital may also be <NAME
cl="CONDITION">infected</NAME>.
While these reports have not
yet been officially confirmed by
national authorities, they appear to
be accurate.
 <NAME cl="ORGANIZATION">WHO</NAME>
is seeking confirmation and
further information from the <NAME
cl="ORGANIZATION">Ministry of
Health</NAME>.  </DOC>
```

Figure 1: Example Annotated Entry from the BioCaster Corpus

a truncated example, and Kawazoe et al. (2006) for a description of the annotation scheme). The corpus consists of around 290,000 words (excluding annotation). Articles were collected from various online news and non-governmental organization sources, including online news from major newswire publishers.[3] Four *per cent* of the corpus was originally gathered by the International Society for Infectious Diseases, under the ProMED-Mail Programme – a human curated disease outbreak report service.[4] From the perspective of the current work, an important characteristic of the corpus is that each document is classified as belonging to one (and only one) relevancy category with respect to infectious disease outbreaks. There are four categories:

[3]Major sources included the BBC (UK), CBC (Canada), *The Nation* (Thailand), IRIN (United Nations), and the *Sydney Morning Herald*, among others.

[4]http://www.promedmail.org

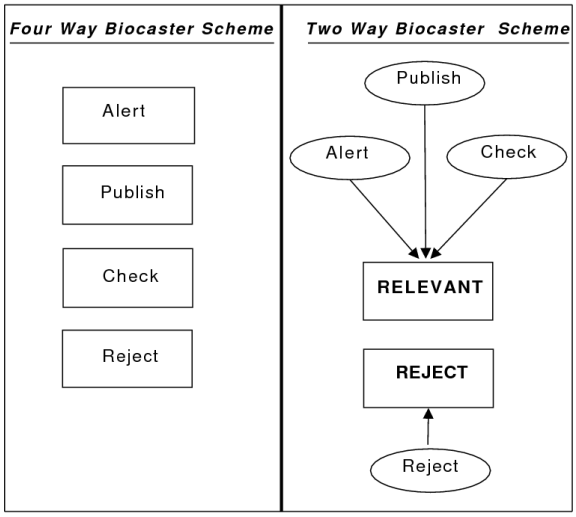


Figure 2: Binary Categories in BioCaster Corpus

- **Alert** — News stories tagged "alert" are deemed to be of immediate interest to health professionals.
- **Publish** — News stories tagged "publish" are judged to be of archival importance to health professionals.
- **Check** — News stories tagged "check" are deemed to be of possible interest to health professionals. The category includes suspicious sounding disease outbreak events for which full information is not available.
- **Reject** — News stories tagged "reject" are deemed to be of little or no interest to health professionals.

In situations where annotators disagreed on the class of a document a domain expert was consulted for clarification. All these categories (and guidelines for determining categories) were developed in consultation with the National Institute of Infectious Diseases (Japan) and based on World Health Organization guidelines.[5]

The corpus is composed of news articles from several different domains (see Table 1). Although over half of the documents in the corpus are classified as belonging to the *health* domain, it is important to stress that articles classified as *alert*, *publish* or *check* can also be found in the *business* category (say, the effect of a livestock disease on the agricultural sector) or in the science and technology category. Additionally, an article may be concerned with a specific infectious disease, but not directly concerned with an *out-*

[5]The WHO guidelines can be found at: www.who.int/gb/ghs/pdf/IHR_IGWG2_ID4-en.pdf

Domain	Number of Documents
Health	539
Business	173
Society	85
Sport	50
Politics	95
ScienceTech	8
Science	44
Technology	3
Entertainment	3

Table 1: Domains in the BioCaster Corpus

break of that disease. Instead, the article could be about a vaccination campaign or a medical breakthrough. Also, the corpus contains documents which are about serious *non*-infectious diseases, like, for instance, most forms of cancer. These non-infectious disease news stories are marked as *reject*.

In order to create a binary classification scheme, the three categories that can broadly be described as relevant with respect to infectious disease outbreaks (*publish*, *alert* and *check*) were conflated into a single *relevant* category (see Figure 2). The binary corpus consists of 350 *relevant* documents and 650 *non-relevant* documents.

Doan et al. (2007), working on an identical task, points out that a bag-of-words representation struggles to identify biomedically relevant senses of polysemous words like *virus* (computer virus or biological virus) or *control* (control a disease outbreak or control inflation) and proposed the use of NE based semantic features as a possible solution.

The approach outlined in this paper develops the work reported in Doan et al. (2007) for binary classification of the BioCaster corpus. We take Doan et al. (2007)'s work one stage further by employing n-grams, a semantic tagger and feature selection to achieve enhanced classification accuracy.

3 Feature Sets

The text classification community has expended a huge amount of research effort on identifying the most effective features for representing text documents. Yet the simplest and most commonly used text representation — the so-called "bag-of-words" representation where each distinct word in a document collection acts as a feature — has proven stubbornly effective. Lewis (1992) compared simple phrase based features with a bag-of-words representation and found that classification performance deteriorated when more complex features were used. The use of syntactic features was again assessed by Moschitti and Basili (2004), who found "overwhelming evidence" that syntactic features fail to improve topic based classification. Scott and Matwin (1999) in a series of experiments using Reuters news wire data reported that phrase based representations (in this case, noun phrases) failed to improve topic classification compared to bag-of-words, and concluded that, "it is probably no worth pursuing simple phrase based representations any further." Domain sensitive *semantic* representations have however been shown to enhance text representations in some situations (Doan et al., 2007).

Named Entity	Attributes
PERSON	case,number
ORGANIZATION	none
LOCATION	none
TIME	none
DISEASE	none
CONDITION	none
NON-HUMAN	transmission
VIRUS	none
OUTBREAK	none
ANATOMY	transmission
SYMPTOM	non
CONTROL	none
CHEMICAL	therapeutic,transmission
DNA	none
RNA	none
PROTEIN	none

Table 2: Named Entities and Roles in the BioCaster Named Entity Annotation Scheme

3.1 Named Entity Based Features

Doan et al. (2007) used the 18 NE tags (some of which have associated attributes or "roles") in the BioCaster annotation scheme to augment bag-of-words features (see Table 2 for a list of NEs and their associated roles), increasing classification accuracy from 74% accuracy with a bag-of-words representation (BOW) to 84.4 % accuracy with a feature set consisting of BOW plus all NS and all NE attributes (BOW+NE+roles). Figure 3 shows how features were generated from a sentence snippet of the BioCaster corpus.

3.2 N-gram Features

N-grams were used (where $n > 1$) as they may help reduce the problems presented by polysemous words and identify concepts highly char-

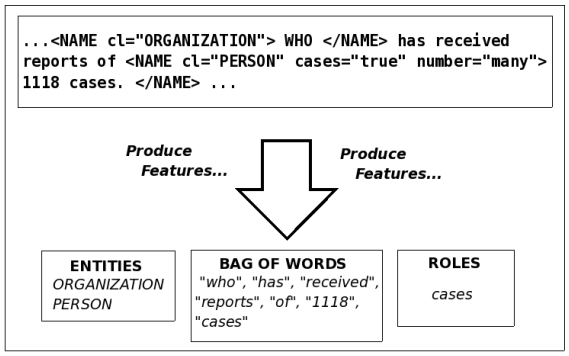


Figure 3: Generating BOW+NE+roles Features (Based on Doan et al. (2007))

acteristic of disease outbreak reports. The trigram `ministry_of_health` may help identify disease outbreak reports more effectively than its constituent unigrams `ministry`, `of` and `health`. Unigrams were derived from the `BioCaster` corpus itself, whereas bigrams and trigrams were acquired from a larger in-domain corpus of 874,000 words from ProMED-Mail disease outbreak report service. This was used in preference to the `BioCaster` corpus because of its size. Only bigrams and trigrams that occurred at least twice in the ProMED-Mail corpus were retained and used in our document representation.

3.3 USAS Semantic Tagger Features

The semantic tags used in this work were generated using the USAS semantic tagger (Rayson et al., 2004).[6] The USAS tag scheme consists of 21 major discourse categories and 232 fine grained semantic tags and relies heavily on a lexicon to assign semantic classes.[7] Figure 4 shows the twenty-one top level categories.

According to Rayson et al. (2004) assigning a semantic tag is a two stage process. First, assigning a list of *possible* semantic tags to a word. Second, identifying the contextually appropriate sense from the list of *possible* tags. A combination of several different methods are used to disambiguate word senses.

- FILTER BY POS TAG. For example, "spring" (season) and "spring" (jump) can be disambiguated using their POS tag. One is a temporal noun and the other is a verb.
- GENERAL LIKELIHOOD RANKING. For example, "green" is used more frequently as a colour term rather than meaning "naïve."
- DOMAIN OF DISCOURSE. The domain of discourse can be specified, and this extra information used in assigning semantic tags. For example, in the food domain, "battered" is more likely to refer to the cooking technique, rather than, say violence.
- TEXT-BASED DISAMBIGUATION. Leverages the fact that a word is likely to retain the same sense throughout a given text.
- CONTEXTUAL RULES. Templates are used to identify some senses. For example, if the noun "account" occurs in the pattern "NP ac-

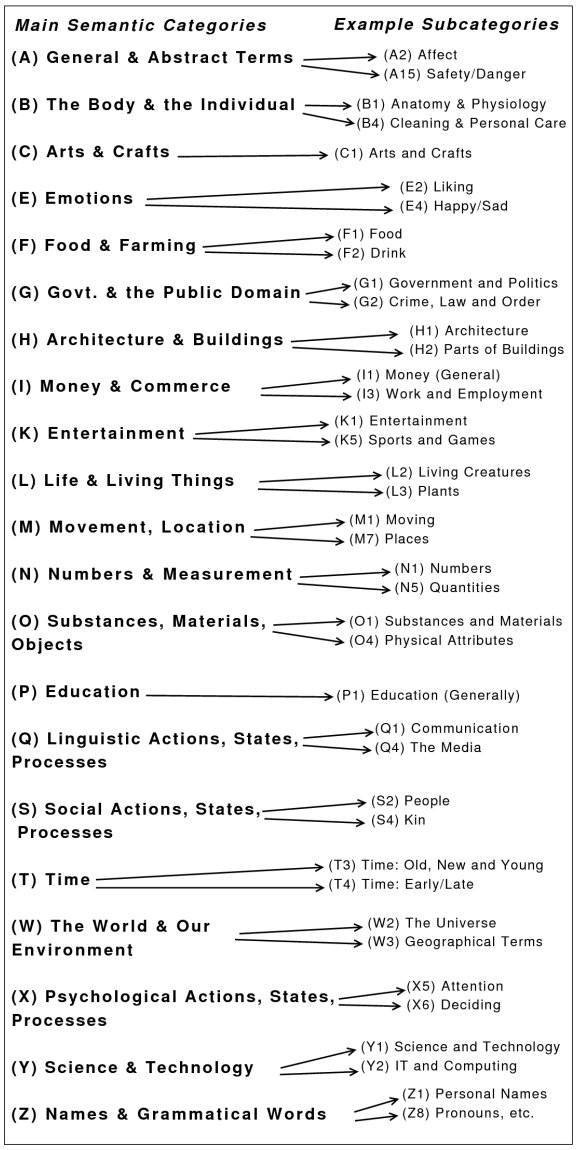


Figure 4: UCREL Semantic Tag Scheme

[6]The USAS (UCREL Semantic Analysis System) was developed at the University Centre for Computer Corpus Research on Language (UCREL) at the University of Lancaster. More details of the tagger can be found at: `http://ucrel.lancs.ac.uk/usas/`

[7]The tagset used in the USAS semantic tagger was loosely based on that developed by McArthur (1981).

count of NP" it is likely to be concerned with narrative explanation.

- LOCAL PROBABILISTIC DISAMBIGUATION. Uses local context and collocational information to determine the correct tag. This method is only partially implemented.

The tagger is also designed to identify multi word units (For example, "United States" is tagged as a multiword unit with a geographical tag) using various techniques, but for the purposes of this work, multiword units were ignored. Also, in some instances the tagger presents two tags as joint equal in likelihood. For example, in the sentence, "County health officials said the baby also exposed about 58 children at the Murray Callan Swim **School**, also in Pacific Beach," the highlighted word "**School**" is classified as both *Education in general* and *Architecture: Kinds of Houses and Buildings*. In this kind of situation – where two tags are presented as equally likely, both tags are retained and used in the document representation.

The tagger has previously been embedded in a translation support system for English and Russian (Sharoff et al., 2006), and has been used in the study of the compositionality of multiword expressions (Piao et al., 2006). An important difference between the USAS semantic tagger and other more well known lexical semantic resources, like WORDNET (Fellbaum, 1998) is that the USAS tagger *disambiguates* between word-senses (albeit without 100% accuracy), rather than providing sets of synonyms for each word sense. Like WORDNET, the USAS semantic tagger is designed for general purpose use, rather than specifically built for use in the biomedical domain.[8] However, 7.7% of words in the taggers lexical database (3,511 words from a total of 45,870) do have *the body* or *life and living things* as their primary semantic category. Table 3 shows a breakdown of the number of words for which a biological sense is dominant.

Tag	Tag Gloss	Lexemes
B1	Anatomy & Physiology	756
B2	Health & Disease	25
B3	Medicines & Medical Treatment	348
L1	Life & Living Things	14
L2	Living Creatures Generally	300
L3	Plants	371

Table 3: Biology Related USAS Semantic Tagger Tags

	REL correct	non-REL correct
Assigned REL	a	b
Assigned non-REL	c	d

Table 4: Contingency Table for Calculating Classification Accuracy (REL is "Relevant" and non-REL is "Non-Relevant")

4 Methodology

In all our experiments, we used a binary feature representation. That is, if a feature X occurs at least once in a document, the feature value for X in that document is 1, otherwise the value is 0. This binary representation was used as early experimental work indicated that binary features performed better than weighted features. Three machine learning algorithms were used: Naïve Bayes, Support Vector machines and the C4.5 decision tree algorithm (Witten and Frank, 2005; Mitchell, 1997). The `Weka` data mining toolkit[9] was used for all the reported machine learning work, and the classification accuracy levels reported (that is, per cent of correctly assigned instances) are the results of 10-fold cross validation. Where statistical significance levels are reported, 10 $\times$ 10-fold cross validation is used in conjunction with the corrected resampled t-test as presented in Bouckaert and Frank (2004). Accuracy is the percentage of correctly defined documents (defined as the number of correctly assigned instances divided by the total number of instances). This can easily be calculated from a contingency table (see Table 4) as $accuracy = (a + d)/(a + b + c + d)$.

Feature selection techniques are central to this work. Yang and Pedersen (1997) showed that aggressive feature selection can increase classification accuracy for certain kinds of texts (in their case, newswire articles). Of the various different algorithms tested, they found that χ^2 and information gain proved most effective. Forman (2003) provides a survey of feature selection methods for text classification.

The χ^2 method was used for feature selection[10]

[8] Note that the general purpose biological categories used by the USAS tagger, while appropriate for disease related newspaper texts in the `BioCaster` corpus, may well be insufficiently fine grained for effectively representing academic papers in the biology domain.

[9] `http://www.cs.waikato.ac.nz/ml/weka/`

[10] The `Weka` implementation of the χ^2 feature selection algorithm was used.

Features	No. Features	NB	SVM	C4.5
semtag	580	78.8	82.8	76.9
semtag (comp)	263	78.4	82.87	74.14
unigrams	21322	88.4	90.9	80.8
bigrams	1567	87.6	87.1	83.5
trigrams	2345	82.5	81.1	82.2
BOW+NE+roles	20889	88.4	90.6	82.2
χ^2 (chi-squared)	9000	**94.8**	92.2	81.6

Table 5: Initial Results (Note that "BOW" is "Bag-of-Words")

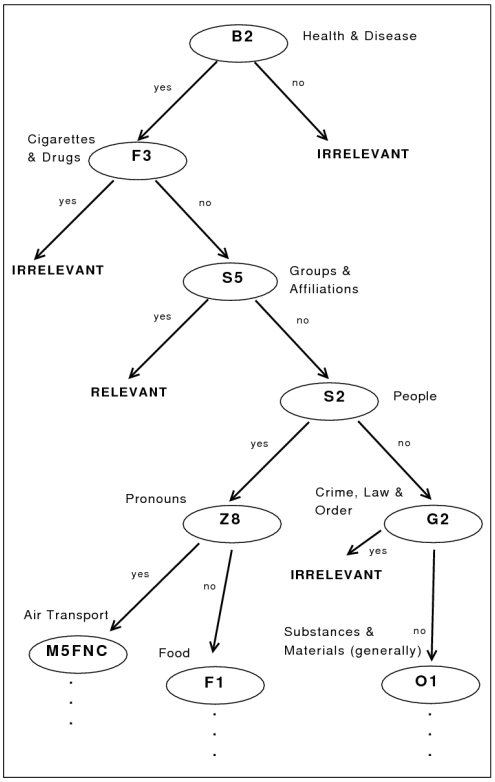


Figure 5: Partial C4.5 Decision Tree for Semantically Tagged `BioCaster` Corpus

as it has shown to be effective in the context of text classification (Yang and Pedersen, 1997). For more on the χ^2 method see Oakes et al. (2001).

5 Results and Discussion

Our chosen baseline in this work is the BOW+NE+roles feature set identified by Doan et al. (2007) using similar data (that is, an earlier, smaller version of the `BioCaster` corpus consisting of 500 documents). This baseline feature set achieved a classification accuracy of 88.4%, the same as the unigram feature set. This was surprising as Doan et al. (2007) found that the BOW+NE+roles achieved higher accuracy than the unigram feature set. These differing results could be accounted for by Doan et al. (2007)'s use of term weighting rather than a binary representation, and the use of a smaller corpus.

Initial comparisons of the several feature representations show that n-gram representations achieved better results than a semantic tag based feature representation. However, a *mixture* of unigrams, bigrams, trigrams and semantic tag features, worked best of all. Table 5 summarizes these initial results. Note that two different document representations based on the USAS semantic tagger were used. The *compressed* representation discarded directionality indicators along a given dimension, and instead used the presence or absence of the dimension itself as a feature. For example, if we take the USAS tag `E2` (Liking), those words tagged `E2+` (like *adore* and *beloved*) and those words tagged `E2-` (like *detest* and *abhor*) will be reduced to one feature (`E2`) reflecting the liking/disliking dimension, although this change had little impact on the results, which are very similar for both of the semantic tagger based representations.

The C4.5 decision tree algorithm seems to perform consistently worse than both the Naïve Bayes and SVM[11] algorithms. One of the advantages of the decision tree algorithm however, is its potential for data exploration purposes. Figure 5 shows the root of a partial decision tree derived from the (full) USAS semantic tag representation of the `BioCaster` corpus. Working from the root of the tree, it can be seen that if the document does not contain any words that are tagged *Health & Disease* then the document is immediately classified as irrelevant (that is, not a disease outbreak report). At the next level, if the document contains a *Cigarettes & Drugs* tag, then the document is classed as irrelevant as diseases *directly* related to cigarettes and non-medicinal drug use are normally chronic rather than highly infectious. The next level down refers to *Groups and Affiliations*, which in the USAS semantic tagger guidelines is described as "Terms relation to groups/the level of association/affiliation between groups,"[12] with prototypical examples like *alliance, caste, community* and so on. The importance of this category for classification accuracy is explained by the inclusion of the word "epidemic" (a strong in-

[11] Default `Weka` parameters were used for the SVM algorithm.

[12] Technical material on the USAS semantic tag scheme can be found at: `http://ucrel.lancs.ac.uk/usas/`

dicator that a document is concerned with disease outbreaks) in the *groups and affiliations* tag.[13]

The best performing representation (94.8% using the Naïve Bayes algorithm – see Table 5) was derived by performing feature selection on *all* the features used (that is, all unigrams, bigrams, trigrams and semantic features). This result was statistically significant when compared to the BOW+NE+roles feature set. Rather than choosing an arbitrary cut off point for feature selection, the optimal number of features was derived experimentally. Figure 6 shows that accuracy peaks at around 9,000 features, and gradually decreases when more features are added.

The 9,000 most powerfully discriminatory features, as determined by the χ^2 method, consist of a mixture of unigrams, bigrams and semantic features, suggesting that a mixed approach to document representation is optimal, rather than relying on a single *type* of feature. Of the one hundred most discriminating features, 50% were unigrams, 37% were bigrams, 8% were trigrams and 5% were semantic tags. As can be seen from Table 6, the two most discriminatory *semantic* features are B2 (health and diseases) and L2 (living creatures), results that are in line with intuitions regarding the subject matter of disease outbreak reports.

Of the 9,000 most discriminating features derived using the χ^2 method, only 130 are semantic tags (<2%), and as semantic tagging is a relatively complex procedure, we investigated the performance of the 9,000 feature set with all 130 semantic features removed, in order to test how much the inclusion of semantic tag features improves accuracy. Running the classifier with the 130 semantic tags removed led to a 0.5% reduction in classification accuracy; not a statistically significant difference.

6 Conclusion

In conclusion, we have shown that for the classification of disease outbreak reports, a combination of bag-of-words, n-grams and semantic features, in conjunction with feature selection, increases classification accuracy at a statistically significant level compared to a "BOW+NE+roles" representation. A novel feature of this work is the use of a semantic tagger — the USAS semantic tagger — to generate semantically rich features. However, most of the increase in classification accuracy arose from the inclusion of n-grams in the feature set, rather than the USAS tagger derived semantic features. It is possible that the thesaurus derived scheme used by the tagger is insufficiently fine grained to capture some important biological concepts, but that the tagger's ability to disambiguate between potentially polysemous biological words (like "virus") was enough to increase accuracy slightly.

Further work will fall into two broad areas:

- Developing and testing further domain specific semantic features (including adding Doan et al. (2007)'s BOW+NE+roles to the feature selection operation).
- Semantic features derived from the USAS tagger will be considered to enhance other

1	health	16	the outbreak
2	cases	17	case
3	outbreak	18	the ministry
4	confirmed	19	hospital
5	died	20	cases of
6	disease	21	poultry
7	symptom	22	outbreak in
8	reported	23	suspected
9	ministry	24	the ministry of
10	death	25	fever
11	virus	26	h5n1
12	the disease	27	have died
13	of health	28	provinces
14	**B2**	29	**L2**
15	ministry of health	30	the virus

Table 6: Most Discriminating Features in the `BioCaster` Corpus

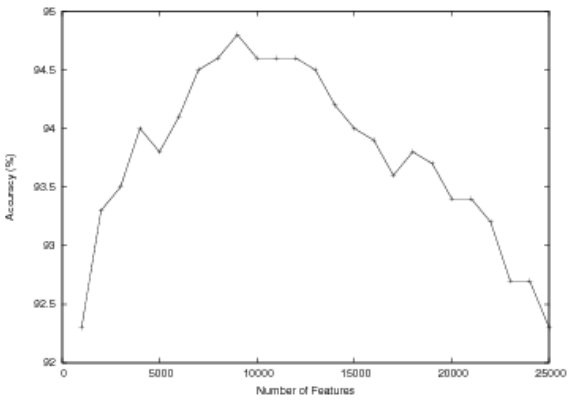


Figure 6: Comparison of Feature Selection Thresholds

[13] As stated above, if the semantic tagger's disambiguation mechanisms cannot decide between two tags, both are included in the document representation. For example, "epidemic" counts as both a *Health and Disease* word, and also as a *Groups and Affiliations* word.

modules of the BioCaster text mining system.

Acknowledgments

We would like to express thanks to Dr Paul Rayson, Directer of UCREL (University Centre for Computer Corpus Research on Language) at Lancaster University for providing access to the USAS semantic tagger. This work was funded in part by grants from the Japanese Society for the Promotion of Science (grant no: P07722) and the Research Organization of Information Systems.

References

R. Bouckaert and E. Frank. 2004. Evaluating the Replicability of Significance Tests for Comparing Learning Algorithms. In *Advances in Knowledge Discovery and Data Mining*, pages 3–12. Springer, Berlin.

A. Cohen and W. Hersh. 2005. A Survey of Current Work in Biomedical Text Mining. *Briefings in Bioinformatics*, 6(1):57–71.

S. Doan, Q. Hung-Ngo, A. Kawazoe, and N. Collier. 2008. Global Health Monitor - A Web Based System for Detecting and Mapping Infectious Diseases. *Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP08) - Companion Volume*, pages 951–956.

S. Doan, A. Kawazoe, and N. Collier. 2007. The Role of Roles in Classifying Annotated Biomedical Text. *BioNLP 2007: A Workshop of ACL 2007*, pages 17–24.

R. Feldman and J. Sanger. 2007. *The Text Mining Handbook: Advanced Approaches to Analyzing Unstructured Data*. CUP.

C. Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, Mass.

George Forman. 2003. An Extensive Empirical Study of Feature Selection Metrics for Text Classification. *Journal of Machine Learning Research*, 3:1289–1305.

D. Heymann, G. Rodier, and WHO. 2001. Hot spots in a wired world: WHO surveillance of emerging and re-emerging infectious diseases. *The Lancet*, 1(5):345-353.

A. Kawazoe, L. Jin, M. Shigematsu, R. Barrero, K. Taniguchi, and N. Collier. 2006. The Development of a Schema for the Annotation of Terms in the BioCaster Disease Detection/Tracking System. In *Proceedings of the Second International Workshop on Formal Biomedical Knowledge Representation*, pages 77–85.

David D. Lewis. 1992. *Representation and Learning in Information Retrieval*. Ph.D. thesis, Department of Computer Science, University of Massachusetts, Amherst, US.

T. McArthur, editor. 1981. *Longman Lexicon of Contemporary English*. Longman, London.

Tom Mitchell. 1997. *Machine Learning*. McGraw-Hill International, Singapore.

A. Moschitti and R. Basili. 2004. Complex Linguistic Features for Text Classification: A Comprehensive Study. In *Proceedings of the 26th European Conference on Information Retrieval Research*, pages 181–196.

M. Oakes, R. Gaizauskas, H. Fowkes, A. Jonsson, V. Wan, and M. Beaulieu. 2001. Comparison Between a Method Based on the Chi-Square Test and a Support Vector Machine for Document Classification. In *Proceedings of the 24th ACM Special Interest Group on Information Retrieval (SIGIR01)*, pages 440–441.

S. Piao, P. Rayson, O. Mudraya, A. Wilson, and R. Garside. 2006. Measuring MWE Compositionality Using Semantic Annotation. *Proceedings of COLING/ACL 2006 Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*, pages 2–11.

P. Rayson, D. Archer, S. Piao, and T. McEnery. 2004. The UCREL Semantic Analysis System. *Proceedings of the Workshop on Beyond Named Entity Recognition: Semantic Labeling for NLP Tasks in association with the Fourth International Conference on Language Resources and Evaluation (LREC 2004)*, pages 7–12.

Sam Scott and Stan Matwin. 1999. Feature Engineering for Text Classification. In *Proceedings of the 16th International Conference on Machine Learning*, pages 379–388.

S. Sharoff, B. Babych, P. Rayson, P. Mudraya, and S. Piao. 2006. ASSIST: Automatic Semantic Assistance for Translators. *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006)*, pages 139–132.

I.H. Witten and E. Frank. 2005. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan-Kaufmann, San Francisco, second edition edition.

Y. Yang and J. Pedersen. 1997. A Comparative Study on Feature Selection in Text Categorization. In *Proceedings of ICML-97, 14th International Conference on Machine Learning*, pages 412–420.

Combining Hidden Markov Models and Latent Semantic Analysis for Topic Segmentation and Labeling: Method and Clinical Application

Filip Ginter,[1] **Hanna Suominen,**[1,2] **Sampo Pyysalo**[2] and **Tapio Salakoski**[1,2]
[1]Department of Information Technology, University of Turku and
[2]Turku Centre for Computer Science (TUCS)
Joukahaisenkatu 3-5,
20520 Turku, Finland
first.last@utu.fi

Abstract

Topic segmentation and labeling systems enable fine-grained information search. However, previously proposed methods require annotated data to adapt to different information needs and have limited applicability to texts with short segment length. We introduce an unsupervised method based on a combination of Hidden Markov Models and latent semantic indexing which allows the topics of interest to be defined freely, without the need for data annotation, and can identify short segments. The method is evaluated in an application domain of intensive care nursing narratives. It is shown to considerably outperform a keyword-based heuristic baseline and to achieve a level of performance comparable to that of a related supervised method trained on 3600 manually annotated words.

1 Introduction

We have previously introduced an application of Hidden Markov Models (HMMs) to topic segmentation (TS) and labeling of Finnish intensive care unit (ICU) nursing narratives (Suominen et al., 2008). In this application, common and repeatedly discussed topics, such as breathing and hemodynamics, are identified in the text, supporting information access and clinical decision-making. In this supervised approach, annotated training data are necessary to induce the HMM model and consequently, the set of possible topics cannot be changed without annotation of additional training data.

In this paper, we introduce a topic segmentation and labeling method where the set of possible topics is not predetermined but is provided by the user as a set of freely chosen keywords, such as *breathing* or *hemodynamics*. The proposed method does not require labeled training data and is, in this respect, unsupervised. This property allows the topics of interest to be easily changed — the user simply specifies new keywords — whereas for a supervised TS and labeling system a new training set would need to be annotated.

The proposed method is a combination of latent semantic analysis (LSA) and a graphical model closely related to HMMs. The method is particularly suitable in cases where almost all documents contain relevant information about the given topics, and the topic segments are short, even shorter than a single sentence. The applicability of existing unsupervised TS methods in these cases is likely to be limited. On the other hand, supervised methods relying on manually labeled training data cannot be applied when the topics can be chosen freely.

Our motivation and scope comes here from ICU narratives. However, we believe that as a general TS and labeling technique supporting *ad hoc* information needs the introduced method may find application also in many other, unrelated domains. As an example of a class of texts which are also characterized by short, unmarked segments, consider scientific publication abstracts, where the method could be applied e.g. to separate between *methods* and *results*-related segments.

2 Related work

TS (alternatively referred to as text segmentation), the automatic division of text into topically coherent units, is a well-studied problem. Many

TS methods are based on the location of first uses of word types, pronoun reference, punctuation marks or other linguistic cues implying topic-change boundaries. The cues are either hard-coded domain-specific rules or induced by machine learning from a corpus (Beeferman et al., 1997; Reynar, 1999).

Another common approach to TS is to consider the similarity of text before and after a proposed segment boundary by measuring, for example, word co-occurrence, repetition or semantic relations; a sudden drop in similarity indicates a likely change in topic. Algorithms based on this approach can be fully unsupervised (Hearst, 1997; Ferret, 2002). Further, LSA has been shown to improve TS when used as a text similarity measure (Bestgen, 2006).

A third major group of TS methods is based on graphical models for sequence labeling. For instance, HMMs have been applied (Yamron et al., 1998; Blei and Moreno, 2001; Suominen et al., 2008). These methods are supervised, but otherwise resemble ours; the approach is a natural choice because segmentation is given by the assigned topic labels.

The applicability of existing TS systems is, however, limited in our case. To allow a free choice of topics of interest, we aim at an unsupervised approach. Further, our data is characterized by very short segment length — several topic changes may occur within a single sentence. Existing unsupervised TS methods require considerably longer segment sizes (see, e.g., (Hearst, 1997; Ferret, 2002)) to reliably detect topic change boundaries. For instance, the TextTiling method of Hearst (1997) searches for topic boundaries between contexts of 200 tokens, whereas the average topic length in our data was only 18 tokens, that is, an order of magnitude shorter. For short texts, techniques similar to query expansion in information extraction and use of likely topic length have been proposed (Ponte and Croft, 1997; Chang and Lee, 2003), but these studies do not, however, consider topic labeling.

In our application domain, Cho et al. (2003) have applied TS and labeling to medical narratives from radiology and urology departments. However, their method relies strongly on hard-coded headlining rules, linguistic cues and lexical patterns seen within training examples. TS techniques have also been designed for the temporal order analysis of medical discharge summaries using a statistical parser to segment the sentences into clauses and two supervised classifiers to predict the segment boundaries and assign for every segment pair their time-wise order (Bramsen et al., 2006). Finally, Hiissa et al. (2007) have introduced a supervised system classifying segments of intensive care patient narratives with respect to topics of *breathing*, *blood circulation*, and *pain*; the segments were, however, created manually.

3 Patient documentation data

The data used in this study consists of nursing notes of 516 adult ICU patients. These Finnish patient-specific records are written during every shift and are mainly used for intra-unit information exchange.

The data set consists of 17140 nursing shifts. We apply a simple domain-adapted tokenizer, obtaining 1.2 million tokens (including punctuation). Each shift thus contains, on average, 73 tokens. The most common topics of the text were *breathing*, *hemodynamics*, *consciousness*, *relatives*, and *diuresis*. Approximately half of the shifts contain explicit topic headings, although these are not standardized and are often misspelled or abbreviated. Additionally, the text is often telegraphic and the vocabulary is highly specialized with a substantial amount of professional terminology, unit-specific documentation practices, and frequent misspellings. Figure 1 illustrates the data.

As test data, we use a manually annotated subset consisting of 402 shifts randomly chosen from the records of first 135 patients by their admission date (Suominen et al., 2008). In the annotation we identify segments belonging to the topics listed above; text not belonging to any of these is assigned the topic *other*. The average length of a topic segment is 18 tokens.

4 Method

We now first recall basic notions of LSA and HMMs and then proceed to introduce the unsupervised TS and labeling method which is based on their combination. The main insight of the proposed method is that the LSA similarity of words to the given topic keywords can be used to replace HMM emission probabilities. Whereas a supervised HMM requires labeled data to estimate the

a)
Night shift
B R E A T H I N G: Doing nicely with the mask. Smallish carbondioxide retention after pain killers, otherwise CO2 < 8.Hourly breathing exercises. Mucus -> wheezing. Able to cough faintly&swallow mucus.
C O N S C I O U S N E S S: Spontaneously awake. DRUGNAME 5mg i.m. After that, was able to nod peacefully. Copes the breathing exercices so-and-so. The strenght in the extremities exept the left hand with bandage are weak.
H e m o d : RRmap staying >65. In a sleep quite low RR.Reduced amount of DRUGNAME and full stop in the small huors. Steady SR.
D I U R E S I S : More DRUGNAME -> Diuresis > 150 ml/h. Fuzzy in the evening.
O T H E R : Son and his wife visiting.Hevay moistening to mouth.
2006-02-01 04:55

b)
Long morning s
After admission fast FA which we treid to invert with electrcity (x3) without result. later FA freq extremely varying and quite economic. After 14 o'clock, pulse occasionally tachycardic, slowed down with DRUGNAME and DRUGNAME infusion (load 150 mg, maintenance 1200 mg/day). Inversion to SR at about 17.30.Hemodynamics quite stable, DRUGNAMEinfusion cont with moder dosage.
Diuresis narrow, morning DRUGNAME.
PCWP highish (21). Adequate CI.
Dr flow normal, narrow.
Forenoon: despite medicatio, tried to breath 'against respirator', which is the reason for relaxation (a couple of times).
Own breatthing started and woke up regardless of sedation & kooperative. With CPAP ok ox and ventilation.
2006-12-11 18:02

Figure 1: Example of Finnish nursing notes translated to English preserving all typing errors and typographical properties. The Finnish originals are not included due to space considerations. Note the topic headings in report *a* with the untypical use of the heading *other* instead of the more common *relatives*. In contrast to *a*, the report *b* does not contain explicit topic headings.

emission probabilities, the unsupervised method only requires a single keyword for each topic.

4.1 Latent semantic analysis

LSA is a commonly applied technique for inducing text similarity measures from co-occurrence statistics in a large, unannotated corpus of text. In our case, we use an LSA-based term-term similarity measure. The standard LSA method based on decomposition of the term-by-document matrix is not applicable because the context in which it measures word co-occurrence is the whole document. In our case, however, the topic keywords occur in the majority of documents — here document refers to a single shift — and, more importantly, different topics tend to co-occur in a single document, therefore not allowing document-level distribution of terms to sufficiently distinguish the various topics. Instead, we apply the Word Space model (Schütze, 1998) which decomposes a term-by-term matrix and only considers word co-occurrence within a fixed context window rather than in the whole document, therefore allowing sub-document distributional properties to be accounted for.

We denote the LSA similarity of word $w_j, j \in \{1, \ldots N_w\}$, to topic $q_i, i \in \{1, \ldots N_q\}$, as $lsa(w_j, q_i)$. Here N_w is the vocabulary size, N_q is the number of possible topics, and q_i is the keyword specified by the user for the respective topic. In our experiments, we use the Finnish equivalents of the keywords *breathing*, *hemodynamics*, *consciousness*, *relative* and *diuresis* to define the five annotated topics. The sixth topic, *other*, is characterized as an LSA query *other NOT breathing NOT hemodynamics NOT consciousness NOT relative NOT diuresis*. The negation operator *NOT* is available in Word Space LSA queries (Widdows and Peters, 2003). The resulting LSA scores are illustrated in Figure 2; they are obtained by first performing LSA on unannotated ICU narrative texts and then calculating the LSA similarity of each vocabulary word with the respective topic keyword (or LSA query with negations). Punctuation, numbers, and small number of extremely common stop-words are excluded from the LSA calculation.

4.2 Hidden Markov Models

We model the problem of segmenting the clinical texts and assigning a topic to each resulting segment as a sequence labeling task. Given an input word sequence $w = (w(1), \ldots, w(T))$, each word $w(t)$, $t \in \{1, \ldots, T\}$, is assigned a topic label $q(t) \in \{q_1, \ldots, q_{N_q}\}$. Each word $w(t)$ belongs to the vocabulary $\{w_1, \ldots, w_{N_w}\}$.

The sequence labeling problem can be solved by an HMM with N_q states where w corresponds to the visible sequence of observations and the sequence of labels $q = (q(1), \ldots, q(T))$ corresponds to the hidden sequence of HMM states. We use a first-order HMM, thus a particular hidden variable $q(t)$ only depends on the previous hidden state $q(t-1)$, and an observed variable

RELATIVES		HEMODYNAMICS		OTHER	
relative	1.000	hemodynamics	1.000	stomach	0.683
phone	0.947	pulse	0.910	other	0.682
daughter	0.916	sr	0.819	net	0.676
wife	0.889	rr-level	0.785	hemolyzed	0.673
visit	0.877	highish	0.784	shirt	0.637
son	0.859	sinus_rythm	0.784	contrast_medium_boosted	0.635
watch	0.821	rr	0.768	blanket	0.630
husband	0.820	blood_pressure	0.716	from_DRUGNAME	0.618
brother	0.785	extrasystole	0.673	soft	0.618
sister	0.777	ok	0.672	puncture_sample	0.614

Figure 2: Translated examples of the words most similar to selected topics and their associated LSA similarity values.

$w(t)$ is only dependent on the value of the hidden variable $q(t)$. Additionally, the initial probability of states is uniformly distributed. The labeling given by the HMM is the best hidden state sequence $\hat{q}$ obtained by solving

$$\hat{q} = \arg\max_{q \in \mathcal{Q}} P(w, q), \qquad (1)$$

where $\mathcal{Q}$ is the space of all hidden state sequences and

$$P(w, q) = P(w(1)|q(1)) \cdot \prod_{t=2}^{T} P(w(t)|q(t))P(q(t)|q(t-1)).$$

The optimal sequence $\hat{q}$ is known as the Viterbi path and the optimization problem (1) can be efficiently computed using the standard Viterbi algorithm. For a detailed introduction to these algorithms, see, for example, Rabiner (1989).

4.3 The proposed unsupervised method

In order to solve (1), the conditional probabilities $P(w(t)|q(t))$, typically referred to as the *emission probabilities*, and $P(q(t)|q(t-1))$, typically referred to as the *transition probabilities*, must be defined. In the supervised case, these are obtained from training data as maximum-likelihood estimates. Here we aim to obtain these conditional probabilities in a minimally-supervised manner which does not require annotated training data. To simplify the notation, we will refer in the following text, whenever possible, to the conditional probabilities $P(w_j|q_i)$ and $P(q_j|q_i)$ without the sequence index t.

4.3.1 Transition probabilities $P(q_j|q_i)$

We distribute the transition probabilities uniformly since, due to our unsupervised setting, there is no annotated data available for direct estimation. In order to be able to control the likelihood of switching from one topic to another, thus controlling the segmentation granularity, we introduce a *self-transition probability* parameter $\delta \in (0, 1)$. The HMM transition probability is then defined as

$$P(q_j|q_i) = \begin{cases} \delta & \text{if } j = i \\ \frac{1-\delta}{N_q - 1} & \text{if } j \neq i \end{cases}.$$

The probability of continuing the current topic is thus δ, and the remaining probability $1 - \delta$ of switching a topic is distributed evenly. Trivially, $\sum_{q_j} P(q_j|q_i) = 1$ for any q_i.

4.3.2 Emission probabilities $P(w_j|q_i)$

Our aim is to derive the value of the emission probability $P(w_j|q_i)$ from the LSA similarity $lsa(w_j, q_i)$ of the word w_j to the topic q_i, or more accurately to the keyword that defines the topic q_i. A straightforward approach is to normalize the LSA similarity into probabilities so that

$$P(w_j|q_i) = \frac{lsa(w_j, q_i)}{\sum_{k=1}^{N_w} lsa(w_k, q_i)}. \qquad (2)$$

This normalization strategy, however, assumes that there is some total mass of relatedness to be redistributed by LSA among the individual words and that this mass is topic-independent. Otherwise, a topic with a small number of related terms will distribute the probability mass of 1 among a small number of words as opposed to a topic with a large number of related terms. Consequently, the emission probabilities of such a topic will numerically dominate the calculation of the Viterbi path $\hat{q}$ and result in poor performance of the model — an effect we have observed in our early experiments. We avoid this type of numerical domination by relaxing the HMM model.

4.3.3 Relaxed graphical model

Instead of normalizing the LSA similarities by Equation 2, we use the unnormalized LSA values directly. This yields a graphical model that preserves the overall structure of an HMM but replaces the emission probabilities with a quantity that is not a probability. The optimal state sequence in this graphical model is then obtained by solving $\arg\max_{q \in \mathcal{Q}} C(w, q)$, where

$$C(w,q) = lsa(w(1), q(1)) \cdot \prod_{t=2}^{T} lsa(w(t), q(t)) P(q(t)|q(t-1)).$$

Replacing the probability $P(w(t)|q(t))$ with the non-probability $lsa(w(t), q(t))$ is the only difference between the HMM cost function $P(w, q)$ and the relaxed model cost function $C(w, q)$. This change does not violate any assumptions in the Viterbi algorithm which thus remains directly applicable to the computation of the optimal sequence of states also in the relaxed model.

This relaxed formalization does not suffer from the problem of a single topic numerically dominating the cost function value and, in our preliminary experiments, resulted in a significant gain in performance. However, a problem of mutual incomparability of the LSA similarity values across topics persists; there is no basis for the implicit assumption that the same LSA similarity value corresponds to the same underlying degree of relatedness, regardless of the topic in question. As an illustrative example of the general problem, let us consider a topic q_1 defined by a single keyword u_1. We then have $lsa(u_1, q_1) = 1$ since the LSA similarity of a word to itself is by definition 1. On the other hand, this does not hold for topics defined by more than one keyword, where the similarity of any of the several defining keywords with the topic is strictly smaller than 1 (except in degenerate cases). Consequently, the same degree of relatedness does not necessarily correspond to the same LSA similarity values across topics. A re-scaling strategy is thus called for which would aim to improve the numerical comparability of the LSA values across topics. We introduce one such possible strategy based on the following insight.

Let us consider words in the descending order by their LSA similarity to a topic q_i and compare for each word its LSA similarity with q_i and the maximum of its LSA similarities with any topic other than q_i (see Figure 3 for illustration). The position in the ordering at which, for the first time, a word has a higher similarity with a topic other than q_i, which we refer to as the *impact index*, naturally divides the ordered list of words into two parts. Words up to the impact index are those that have a high LSA similarity to the topic q_i and, at the same time, do not have higher similarity with any other topic. These words are thus strong indicators of the topic q_i. The LSA similarity of the word at the impact index, which we refer to as the *impact similarity* is then, for the topic q_i, a natural cut-off point that gives the lowest LSA similarity at which the words can yet be considered as strong indicators of the topic. Numerically, the impact index and impact index similarity may vary significantly across topics.

Since the impact similarity has a clear intuitive interpretation, we propose a strategy which re-scales the LSA values for each topic so that the impact similarity is set to a given, topic-independent constant α. Additionally, the re-scaling sets the LSA similarity of the most similar word for any topic as equal to 1 and the minimal similarity of any word to any topic to be a constant β. The effect of this re-scaling is illustrated in Figure 3.

We now proceed to define the re-scaling strategy formally. Let us consider an ordering π_i of the words such that the value $\pi_i(w_j)$ gives the index at which the word w_j is found in a sequence of words ordered in descending order by their LSA similarity with q_i. Let $lsa_1(q_i) = \max_{w_j} lsa(w_j, q_i)$ and $lsa_m(q_i) = \min_{w_j} lsa(w_j, q_i)$. Finally, let $lsa_I(q_i)$ denote the LSA similarity $lsa(w_j, q_i)$ where $\pi_i(w_j) = I(q_i)$, that is, the impact point similarity for topic q_i. These quantities are illustrated in Figure 3. The re-scaled LSA similarity, denoted $\overline{lsa}$, is then defined in Equation 3.

The optimal state sequence through our final model is then obtained by solving $\arg\max_{q \in \mathcal{Q}} \overline{C}(w, q)$, where

$$\overline{C}(w,q) = \overline{lsa}(w(1), q(1)) \cdot \prod_{t=2}^{T} \overline{lsa}(w(t), q(t)) P(q(t)|q(t-1)).$$

To summarize, we have now obtained a graphical model for unsupervised topic segmentation and labeling of text that is closely related to first-order HMMs. The transition probabilities other

$$\overline{lsa}(w_j, q_i) = \begin{cases} \frac{1-\alpha}{lsa_1(q_i)-lsa_I(q_i)} \cdot (lsa(w_j, q_i) - lsa_I(q_i)) + \alpha & \text{if } \pi_i(w_j) \leq I(q_i) \\ \frac{\alpha-\beta}{lsa_I(q_i)-lsa_m(q_i)} \cdot (lsa(w_j, q_i) - lsa_m(q_i)) + \beta & \text{otherwise} \end{cases} \quad (3)$$

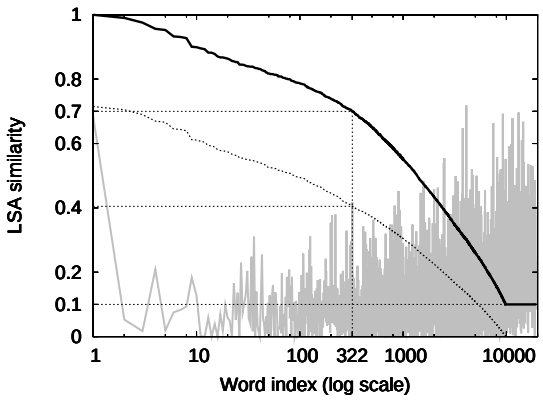


Figure 3: The effects of re-scaling the LSA similarity values of the topic *other* by Equation 3. The re-scaled LSA values are shown as a full line, the unscaled LSA values as a dotted line, and the maximum LSA similarity with any other topic as a gray line. The important characteristics of the LSA values in this case are: $lsa_1(q_i) = 0.71$, $I(q_i) = 322$, $lsa_I(q_i) = 0.41$, and $lsa_m(q_i) = 0$. The re-scaling parameters are $\alpha = 0.7$ and $\beta = 0.1$.

than the parameterized self-transition probability δ are uniformly distributed and the emission probabilities are replaced by LSA similarity values that have been re-scaled to improve numerical comparability across topics. The main difference of this model and the standard supervised HMM is that the proposed model does not require labeled training data. Instead, it only requires a set of keywords defining the topics and large-enough body of unannotated text on which the LSA is calculated. The model is decoded using the standard Viterbi algorithm.

5 Performance evaluation

We evaluate the proposed method on manually annotated gold-standard data (see Section 3). The test set consists of 204 and the training set of 198 annotated shifts randomly selected from 135 patient reports. If two shifts report on the same patient, both are placed either in the train set or in the test set. LSA is calculated from all text available in the 448 patient reports from which no shift was selected into the test set.

To reduce sparseness problems due to the highly-inflective nature of Finnish, we lemmatize the text using a version of the FinTWOL Finnish morphological analyzer[1] (Koskenniemi, 1983) whose lexicon has been extended by approximately 3500 clinical domain terms. For every word analyzed by FinTWOL, we use the first lemma given, and for words outside of FinTWOL lexicon, we use the unchanged surface word form. The LSA similarity scores are computed using the Infomap NLP software[2] (Dorow and Widdows, 2003).

Since a fully-unsupervised parameter-selection method is so-far not available, we select the parameters by grid search on a held-out set of 60 annotated shifts. These shifts are not part of the test set in order to avoid overfitting the parameter selection. The context window width is set to 30 words (left and right context both 15 words), and the method parameters are $\delta = 0.6$, $\alpha = 0.3$, and $\beta = 0.15$. All other LSA-related parameters (max number of singular values, number of Word Space columns, etc.) are left at their default after preliminary experiments indicated that they have only marginal effect on the overall performance.

To establish the relative merit of the unsupervised method, we compared its performance against two other methods: a keyword-trigger method and a comparable supervised learning method. The keyword method is a simple baseline that performs segmentation and labeling by looking for the occurrence of the five topic keywords (*breathing* etc.), assigning each word to a labeled segment corresponding to the previous seen keyword. The assigned label is given the initial value *other* at the start of each shift. To allow the keyword-based approach to benefit from the normalizing effect of morphological analysis, the trigger words are matched against the lemmas given by FinTWOL.

The supervised method compared to is a basic first-order HMM. This choice is made not out of ignorance of advances such as conditional random fields (see, e.g., (Sutton et al., 2007)), but rather as HMM is a close supervised equivalent of the proposed model — we sought to determine

[1] `http://www.lingsoft.fi/`

[2] `http://infomap-nlp.sourceforge.net/`

	Accuracy	WindowDiff
majority baseline	23.4	0.32
keyword baseline	66.9	0.16
unsupervised model	74.9	0.23
supervised HMM	82.9	0.21

Table 1: Performance of the three compared methods. Note that for WindowDiff lower value indicates better performance — a perfect segmentation obtains WindowDiff score of zero. Majority baseline refers to assigning the most common topic in the data (*consciousness*) to all tokens.

the relative efficiency of the unsupervised and supervised alternatives in setting the parameters of the graphical model. For the HMM, the only parameter, the smoothing model and its setting, was selected on the training set by a separate search of the parameter space so as to avoid overfitting the test set. The selected optimal smoothing model was Lidstone (add-γ) smoothing with $\gamma = 0.3$.

The primary evaluation measure is micro-averaged accuracy, the proportion of words in the test set with correctly identified label. Further, we report macro-averaged WindowDiff (Pevzner and Hearst, 2002) score, which is often used to evaluate segmentation quality independently of the topic labels. The WindowDiff window size was set to half of the average segment size in the gold standard data, a standard way to set this parameter. Note that WindowDiff only takes into account the positions of segment boundaries, ignoring the topic labels.

6 Results and discussion

The performance of the methods on the test set (204 shifts, 15839 tokens) is reported in Table 1. As expected, the accuracy of the unsupervised model is between the performance of the keyword baseline and the supervised HMM. The unsupervised model considerably outperforms the keyword baseline. Further, it is not surprising that the supervised HMM performs better than the unsupervised model, considering that it receives much more detailed information about the distribution of words with respect to topics.

Interestingly, the WindowDiff results are in disagreement with the accuracy results, with the keyword baseline reaching better WindowDiff performance than even the supervised HMM. We have currently no explanation for this highly unintuitive secondary result. Nevertheless, as the unsupervised method performs nearly at the level of the supervised in terms of WindowDiff and this measure does not take into account the assigned labels, a key aspect of the method, we do not view this result as compromising the positive primary findings in terms of accuracy. In the following, we will only focus on accuracy results.

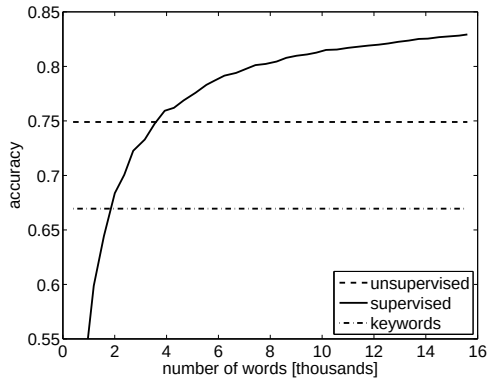


Figure 4: Learning curve for the supervised baseline method. The performance of the unsupervised and keyword-based methods are shown for reference.

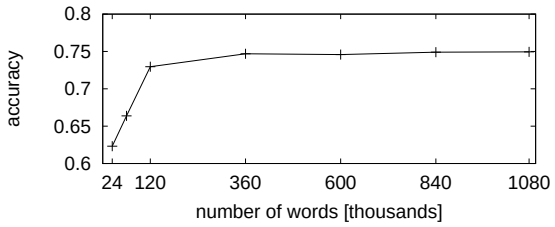


Figure 5: Learning curve for the unsupervised method.

An interesting question to ask here is how many words of labeled data does the HMM require to reach the accuracy of the unsupervised method. The learning curve of the HMM, that is the dependence of its accuracy on the amount of available training data, is given in Figure 4. Here we observe that in order to reach the performance of the unsupervised method, it is necessary to manually label roughly 3600 words. For comparison, the learning curve for the unsupervised method is shown in Figure 5; the curve is generated by varying the amount of text available to calculate the LSA. Here we see that the peak performance is reached after about 360,000 words (150 full patient reports). Note that for the unsupervised method the text is not manually labeled; gathering the amount of data necessary for reaching the peak performance does not involve any manual annotation effort, unlike in the case of the supervised HMM.

7 Conclusions and future work

We have introduced an unsupervised method for TS and labeling based on a combination HMMs and LSA. We have shown that, in order to reach the performance of the unsupervised method, a standard HMM would require 3600 words of labeled training data, as opposed to just one keyword per topic necessary for the unsupervised method. The proposed method is thus applicable to information search tasks with freely-chosen topics and no labeled data available. We have applied the method to a real-life clinical task.

In further research, several crucial questions will be investigated. First is that of unsupervised selection of the parameters of the system (such as the LSA window width and self-transition probability δ). The second open question is whether the current proposed model can be re-normalized to obtain an actual HMM without loss of performance. This would open further interesting directions such as the possibility to use the LSA-based HMM model as an initial state for further unsupervised training of the method, for instance by the standard Baum-Welch algorithm. Finally, a general way of modeling the topic *other* is needed for applications where some segments do not belong to any keyword-defined topic.

Acknowledgments

This work was supported by the Academy of Finland and the Finnish Funding Agency for Technology and Innovation, Tekes. We thank Simo Vihjanen and Sari Ahonen from Lingsoft Inc. for extending the FinTWOL lexicon and Heljä Lundgrén-Laine and Päivi Haltia for their advise regarding ICU practices and language.

References

D Beeferman, A Berger, and J Lafferty. 1997. Text segmentation using exponential models. In *Proceedings of EMNLP-2*, pages 35–46. ACL.

Y Bestgen. 2006. Improving text segmentation using Latent Semantic Analysis: A reanalysis of Choi, Wiemer-Hastings, and Moore (2001). *Computational Linguistics*, 32(1):5–12.

DM Blei and PJ Moreno. 2001. Topic segmentation with an aspect hidden Markov model. In *Proceedings of SIGIR'01*, pages 343–348. ACM.

P Bramsen, P Deshpande, YK Lee, and R Barzilay. 2006. Finding temporal order in discharge summaries. In *AMIA Annu Symp Proc 2006*, pages 81–85. AMIA.

T-H Chang and Ch-H Lee. 2003. Topic segmentation for short texts. In *Proceedings of PACLIC 17*, pages 159–165. Colips Publications.

PS Cho, RK Taira, and H Kangarloo. 2003. Automatic section segmentation of medical reports. In *AMIA Annu Symp Proc 2003*, pages 155–159. AMIA.

B Dorow and D Widdows. 2003. Discovering corpus-specific word senses. In *Proceedings of EACL'03*, pages 79–82. ACL.

O Ferret. 2002. Using collocations for topic segmentation and link detection. In *Proceedings of COLING'02*, pages 1–7. ACL.

MA Hearst. 1997. TextTiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64.

M Hiissa, T Pahikkala, H Suominen, T Lehtikunnas, B Back, H Karsten, S Salanterä, and T Salakoski. 2007. Towards automated classification of intensive care nursing narratives. *Int J Med Inform*, 76(S3):362–368.

K Koskenniemi. 1983. Two-level model for morphological analysis. In *Proceedings of IJCAI'83*, pages 683–685. Morgan Kaufmann.

L Pevzner and MA Hearst. 2002. A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics*, 28(1):19–36.

JM Ponte and WB Croft. 1997. Text segmentation by topic. In *Proceedings of ECDL '97*, pages 113–125. Springer-Verlag.

LR Rabiner. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

JC Reynar. 1999. Statistical models for topic segmentation. In *Proceedings of ACL'99*, pages 357–364. ACL.

H Schütze. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–123.

H Suominen, S Pyysalo, F Ginter, and T Salakoski. 2008. Automated text segmentation and topic labeling of clinical narratives. In *Proceedings of Louhi'08*. TUCS. To appear.

C Sutton, A McCallum, and K Rohanimanesh. 2007. Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data. *J Mach Learn Res*, 8:693–723.

D Widdows and S Peters. 2003. Word vectors and quantum logic: Experiments with negation and disjunction. In *Proceedings of MoL8*, pages 141–154.

JP Yamron, I Carp, L Gillick, S Lowe, and P van Mulbregt. 1998. A hidden Markov model approach to text segmentation and event tracking. In *Proceedings of ICASSP'98*, pages 333–336. IEEE.

Complex-to-Pairwise Mapping of Biological Relationships using a Semantic Network Representation

Juho Heimonen,[1] **Sampo Pyysalo,**[2] **Filip Ginter**[1] and **Tapio Salakoski**[1,2]
[1]Department of Information Technology, University of Turku
[2]Turku Centre for Computer Science (TUCS)
Joukahaisenkatu 3–5
20520 Turku, Finland
first.last@utu.fi

Abstract

This study examines representations of protein–protein interactions focusing on the mapping between simple, pairwise annotation and complex, structured annotation. A simple semantic network representation equivalent to the BioInfer predicate formalism is introduced and used to transform the complex annotation of BioInfer into pairwise annotation through hand-written rules. Evaluation shows that this binarisation can be largely validly performed with limited loss of information, but also reveals specific challenges. The binarised BioInfer is the first corpus of this type where the inclusion rules are formalised to the level of a computational implementation and is freely available at http://www.it.utu.fi/BioInfer.

1 Introduction

The identification of protein-protein interactions (PPI) from free text is one of the most important and widely studied information extraction tasks in biomedical natural language processing. Automatic PPI extraction would benefit a wide range of applications, from advanced search engines to automated pathway database construction.

The great majority of PPI extraction methods and annotated corpora have cast the task as one of identifying pairs of protein names for which some relationship is stated. While the simplest case of extracting unordered pairs is the most widely studied, approaches targeting e.g. ordered pairs or pairs with a connecting relationship type (e.g. Ding et al. (2002), Nédellec (2005)) have also been published, as have some methods for extracting n-ary (for $n > 2$) relations (McDonald et al., 2005). However, pairwise approaches remain the norm and the information extracted by these constitutes only a small part of the knowledge in biomedical literature.

Recently two corpora that contain PPI annotation considerably more detailed than pairwise relations have been introduced. These resources, the BioInfer (Pyysalo et al., 2007) and GENIA Event (Kim et al., 2008) corpora, aid the development of extraction systems that capture complex PPI—here, understood to refer to n-ary interactions of proteins and to include also structured (nested) relations where, for example, a protein affects the interaction of other proteins. This paper explores the relationship between this type of complex annotation and the prevailing pairwise annotation.

First, it is argued that a representation capable of capturing the core of information in complex relationships while remaining practical to extract is needed in complex PPI extraction. In this paper, protein relationships are represented as semantic networks. Since they are based on the BioInfer annotation, these networks follow the textual expressions of the statements of those relationships and are capable of expressing complex PPI. While this is not a fully formal knowledge representation, it aims to support automatic, consistent derivation of simpler, more easily extracted targets and serve as a practical intermediate between textual expressions and formal biological knowledge.

Second, the representation is applied together with a transformation ruleset tailored for the task of transforming the complex relationships in the BioInfer corpus into typed binary (i.e. pairwise) relationships where the types preserve considerably more information regarding the nature of

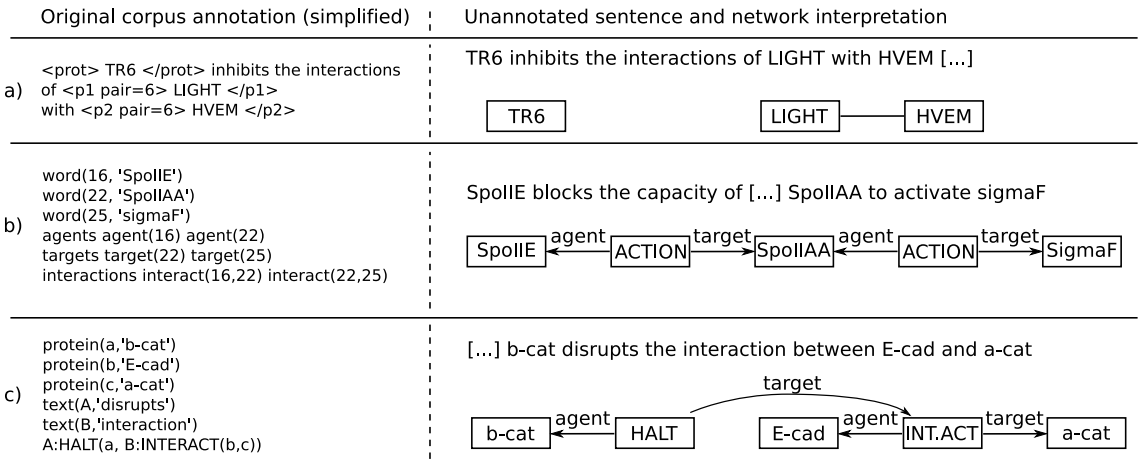


Figure 1: Examples of annotation in a) AIMed, b) LLL and c) BioInfer as semantic networks. Note that the original annotation does not include this representation.

PPI than simple protein pairs. This transformation aims to capture all (and only) biologically meaningful relationships in the original annotation. The transformation is evaluated in a detailed analysis where the magnitude and properties of the information loss necessarily entailed by such a simplification is further discussed along with its significance to the PPI extraction task.

2 Representing biomedical knowledge

The PPI annotation schemes in most domain corpora aim at capturing simple facts about proteins rather than serving as a knowledge representation in the sense of a computable model that supports deductive inference.

Figure 1 illustrates the information contents of annotations in the AIMed (Bunescu et al., 2005), LLL (Nédellec, 2005) and BioInfer corpora as informal semantic networks[1]. AIMed and LLL model interactions as pairwise relationships while BioInfer allows complex relationships. Furthermore, AIMed is not annotated for direction or type while LLL and BioInfer are. The key limitation of pairwise relationship annotation is its incapability to express complex structured relationships. Thus, the annotation involves decomposition that leads to approximations and loss of information. For example, in the LLL annotation in Figure 1b, the effect of SpoIIE on sigmaF is not explicitly annotated and cannot be inferred from the annotation shown in the figure, which is indistinguishable from the annotation that would be given, for example, to *SpoIIE activates SpoIIAA which binds SigmaF*.

In addition to loss of information, the decomposition can lead to inconsistencies. There is large variation in annotation principles (see e.g. Pyysalo et al. (2008)) which evidently leads to annotation of a variety of interaction types across domain resources. For individual corpus annotation efforts, inconsistencies in decomposition principles may contribute to low inter-annotator agreement (see e.g. Alex et al. (2008)).

Despite the limitations of pairwise annotation, pairwise relationships may be necessary in applications such as querying for interactions between two proteins. Assuming that complex relationships are a useful target for information extraction efforts and that simple relationships have benefits in post-extraction applications, a mapping from complex to simple relationships is needed. Further, significant challenges still remain even in pairwise PPI extraction (Krallinger et al., 2007), and while carefully hand-crafted systems extracting complex PPI have been introduced (Friedman et al., 2001), reliable machine-learning approaches to complex PPI extraction may not emerge in the near future. A reliable mapping of the BioInfer and GENIA annotations to pairwise annotations would thus serve to increase the applicability of these resources to presently available extraction methods.

[1]Note that not all the information in Figure 1 is explicitly represented in the corpora: for example, interaction types in LLL are found as comments in the corpus file.

3 Methods and resources

3.1 Corpora

BioInfer was the first domain corpus to introduce the annotation of complex protein relationships. It consists of 1100 sentences annotated for protein names, their relationships, and dependency syntax and uses a predicate formalism in its PPI annotation (see Figure 1c). The GENIA event corpus contains similar annotation, but its relationship annotation of 1000 PubMed abstracts was published late during the present study, which thus focuses on the BioInfer corpus. The essential features of the PPI annotation of the BioInfer and GENIA corpora are largely identical: complex relationships are annotated, participants in relationships are not restricted to protein names but refer to the actual participants even when these are e.g. abstract entities such as *gene expression*, and the annotation is fully bound to the text. Therefore, the methods described in this paper could well be applied to GENIA in a future study.

3.2 Semantic network representation

The term *semantic network* can refer to a variety of graphical representations of knowledge which differ in expressive strength and complexity. A graph representation is a natural choice for semantics, and several well-developed and powerful formalisms have been introduced (Sowa, 1976; Mel'čuk, 1988). However, their complexity makes them difficult targets for automated extraction. An ideal representation for PPI extraction would be as simple as possible, yet capable of capturing all PPI statement types in natural language, and formally well-founded.

In the context of this paper, a semantic network is understood to refer to a directed graph in which the nodes represent biological concepts and the edges represent the stated roles of these concepts. As the applied networks derive from the BioInfer predicate annotation, the graphs are further acyclic, that is, DAGs. The nodes are bound to their corresponding textual expressions through *text bindings* following the original BioInfer annotation. A relationship is defined as a directed subtree with at least two leaves, and a relationship composed of an entire subtree rooted at a source (DAG "root") is termed a complete relationship. In this model a binary relationship is defined as a relationship containing exactly three nodes, two of which are leaves, and a complex relationship is one that is not binary.

type	meaning
agent	agent in an asymmetric process
patient	patient in an asymmetric process
participant	participant in a symmetric process
sub	substructure or member
super	superstructure, family or group
identity	identical entities
possessor	possessor of a property

Table 1: Edge types used in the semantic network.

The nodes and the edges in the network can represent any concept of interest and any semantically sound role, respectively. However, the set of valid edge types is restricted by the type of the predecessor. For example, *actin* (a physical entity) can have an agent or patient role in *depolymerisation* (a process) but not in *filaments* (another physical entity). A controlled vocabulary or, ideally, an ontology must be employed to accurately and formally express the knowledge.

A predicate representation such as that of BioInfer can be directly mapped into an equivalent semantic network where the node types correspond to predicates and their arguments and the edge types only distinguish between the argument positions (1^{st}, 2^{nd} etc.). In case of BioInfer, the node types thus correspond to types in the BioInfer ontologies. Further, edge types (shown in Table 1) are indirectly obtained from the description of the nesting and the predicates (see Section 3.3.1). Thus, the network representation can capture the same general set of biomedical relationships as the original BioInfer annotation. However, the network representation has several practical advantages over the predicate representation of BioInfer. Biological concepts, which can be either physical, such as molecules or cell components, or abstract, such as processes, properties or relationships, are represented in a unified manner, unlike in the predicate representation that differentiates between predicates (relationships) and entities. Further, the participant roles are explicitly represented, facilitating processing of relationships. Finally, the network representation is naturally extensible: for example, information regarding cell type could be added simply by attaching additional edges to the network.

Figure 2 provides an example of a semantic network that uses the BioInfer ontologies.

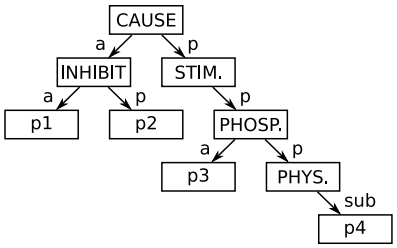


Figure 2: Example of a semantic network representing the sentence *Inhibition of B by A causes stimulation of phosphorylation of D filaments by C. Agent* is abbreviated as *a* and *patient* as *p*.

The fact that no agent is stated for the node *STIM.* (STIMULATE) renders this particular relationship unexpressable in the BioInfer formalism without adding an anonymous entity.

3.3 Binarisation process

Here, binarisation is defined as a process of mapping a complex relationship into a set of (typed) binary relationships, aiming at sound (valid, truth-preserving) inference as well as to preserve the key biological information of the original relationship. This is achieved through a corpus-specific set of hand-written inference rules. Instead of formal inference (as understood in logic) aiming at finding new (unstated) knowledge, the purpose of the inference rules is to reduce original annotation into binary annotation by applying transformations that generate the most accurate approximation of the original information content.

The validity of inference is evaluated with respect to biologists' understanding of whether the generated binary relationships describe relations stated in the text. Ideally, the binary annotation includes all (and only) pairwise PPI that are biologically relevant, along with appropriate types. Note that not all protein pairs forming a relationship generate biologically relevant binary relationships: for example, no such relationship can be validly inferred between p_1 and p_3 from the statement p_1 *prevents the phosphorylation of* p_2 *by* p_3. By contrast, for p_1 *prevents the binding of* p_2 *to* p_3, a p_1–p_3 relationship could be inferred because *bind* is a symmetric relationship.

Before binarisation, the semantic network is preprocessed to simplify the binarisation process and to separate the binarisation from refinement of relationships.

3.3.1 Preprocessing of the network

The BioInfer corpus contains annotation for a number of non-biological relationship types, such as equality and coreference, which are used to detail the expression of other, biological, relationships. Non-biological relationships are excluded from the binarised corpus. However, to preserve as much biological information as possible, these relationships are resolved by graph transformations following their interpretations, as given in (Pyysalo et al., 2007).

For example, in BioInfer the EQUAL predicate is used to express identity relationships, mostly in abbreviations and synonym definitions, and the COREFER predicate is used to express coreference. Only the first argument of these predicates is then used in other relationships, and thus in the network these relationships are introduced for the second argument by copying edges and nodes, as illustrated in Figure 3.

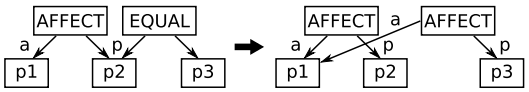


Figure 3: Preprocessing EQUAL predicates. The annotation AFFECT($p1$,$p2$) EQUAL($p2$,$p3$) for the expression $p1$ *affects* $p2$ *(also called* $p3$*)* is preprocessed into AFFECT($p1$,$p2$) AFFECT($p1$,$p3$).

In the BioInfer entity annotation, entities can be nested, i.e. contain other entities: for example, *p1 subunit* is annotated as two entities, *p1 subunit* and the nested *p1*. However, the annotation does not specify the type of the relations implied by nesting. These relations are represented as edges in the network and their types can be resolved reliably by heuristics based on the types and text bindings of the end nodes of the edges. For example, in *[depolymerisation of [[actin] filaments]]* the edge from *depolymerisation of* to *filaments* is resolved into *patient* (rule: physical entity nested in a process with *of* in its text binding) and the edge from *filaments* to *actin* is resolved into *sub* (rule: physical entity nested in larger physical entity). The special predicate REL-ENT, implying indirect nesting, is resolved similarly.

3.3.2 Extraction of binary relationships

Binary relationships are extracted in a two-step process. First, candidate relationships are generated from the original graph by forming all possible relationships with exactly two proteins

as leaves. In order to determine the polarity of the resulting binary relationship, all adjacent nodes of type NOT are included into the relationship. Since the edges are explicitly labeled with roles whose interpretation is independent of other edges, such a subgraph is sufficient to preserve all the details of the relationship between the two selected proteins while being easier to process than the entire graph.

Second, the relationships are transformed with a set of rules that reduce them into binary relationships. Each rule defines a transformation that aims to preserve the information content while simplifying the relationship by removing nodes and/or altering the types of the nodes and edges. Unlike in formal inference, each transformation produces an approximated relationship, and the validity of the inference is not guaranteed. To minimise the overall extent of approximations and to avoid invalid inference, the rules are manually ordered so that more reliable and less approximative rules have priority.

Rules including the root determine the final relationship type and are applied first. Essentially these rules process nodes representing verbs with little semantic content as well as determine the overall regulatory effect. Rules applying to leaves remove nodes whose information content cannot be included in the final relationship, and are applied only if other rules do not match. In most cases, the removed information concerns the details of the exact types of the physical entities. By iteratively applying the first matching rule, each relationship is transformed until a binary relationship is obtained or none of the rules match. The semantic network representing all valid binary relationships is simply the union of the binary relationships obtained in this step.

Figure 4 illustrates the transformation process. In step a), a node representing the verb *cause* is removed. This is a minor approximation since the node (*CAUSE*) indicates that *p1* is (indirectly) an agent in the stimulation process. Similarly, an agent of a regulatory process (*INHIBIT*) causing another process (*STIM.*) is indirectly the agent of that other process. Hence, *INHIBIT* is removed in step b). Step c) is a rearrangement of nodes: a regulatory process (*STIM.*) is processed into the effect attribute (see Section 4.1) of the affected physical process (*PHOS.*). In step d), it is approximated that anything that is stated for a physical entity (*PHYS.*) is also valid for its component (*p4*). In this example the resulting relationship is REG(+)_PHOSPHORYLATE(p1,p4).

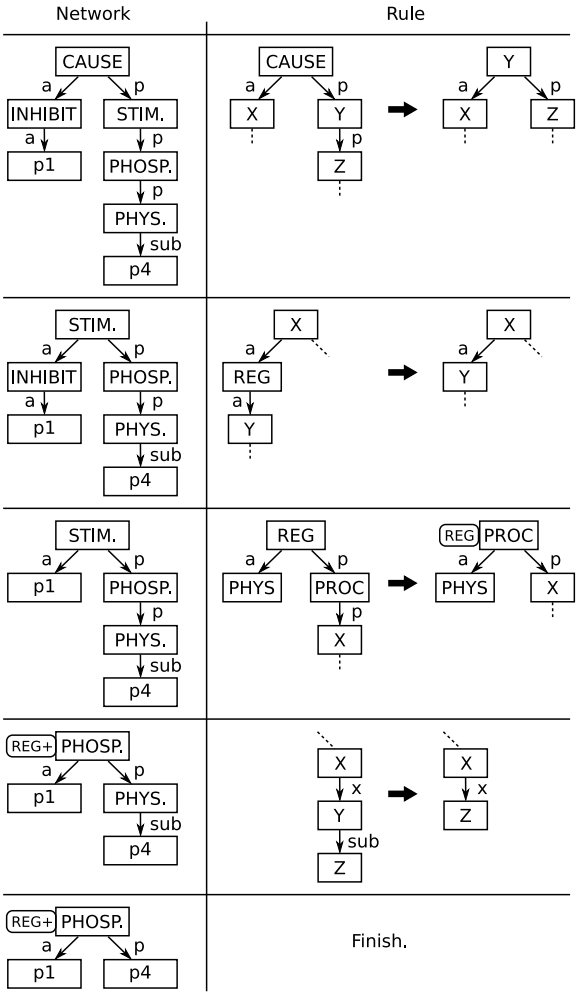


Figure 4: An example of candidate relationship processing. See Figure 2 for description of notation and Section 4.1 for REG(+) attribute description.

3.4 Development and evaluation protocol

In order to be able to fairly evaluate the effect of the binarisation process on previously-unseen data, the software and rules were developed on a random sample of 437 sentences. The process was then applied to the complete BioInfer corpus and all relations in a random sample of 50 previously-unseen sentences of the binarised BioInfer were analysed by a biologist to determine the quality of the binarisation.

In the error analysis, instances of information loss were counted and their causes examined. The losses were categorised as follows, in decreasing order of severity: missing interaction, invalid inference, invalid interaction text binding,

approximated interaction type, and lost interaction detail. The latter two were considered as approximations while the other as errors. The lost interaction details were divided into three categories (process/property, structure/membership, identity) and evaluated by counting the entities that did not contribute to the corresponding binarised relationship.

The applied proof-of-concept software is implemented in Python and Prolog. Any similar programming language or inference tool would be equally good provided that it supports the ordering of the rules and the search for the first sequence (based on the rule order) of transformations leading to a binary relationship.

4 Results and discussion

4.1 Binarisation details

Single BioInfer predicate types are not alone sufficient to summarise complex relationships. In particular, polarity needs to be preserved to separate explicit negative statements, originally annotated with the NOT predicate, from unannotated (i.e. non-existing) statements (see Pyysalo et al. (2007)). In addition, complex relationships can combine aspects of regulation to the primary effect: for example, the annotation for p_1 *suppresses the polymerisation of* p_2 includes both the SUPPRESS and POLYMERIZE types but neither alone is sufficient to express the whole relationship. To make it possible to preserve negation and regulatory aspects, the predicates are augmented with *polarity* and *effect* attributes.

The base predicate specifies the relevant biological process while the effect attribute describes how this process is affected by the agent. The effect can be positive, negative, or unspecified regulation or a direct action. For simplicity, when polarity or effect have their "default" values (positive and direct action, respectively) these are omitted from the augmented predicate: thus, instead of POS_DIRECT_INHIBIT simply INHIBIT is used as the name. Hence, for example, NEG_POLYMERIZE indicates the agent does not polymerise the patient, REG(-)_BIND indicates that the agent negatively regulates the binding of the patient (to an unspecified entity).

The BioInfer ontologies are modified to better support the binarisation as follows. The Process_entity subtree in the entity ontology is mapped to the relationship ontology: for example, the process entity DEPOLYMERIZATION is mapped to the predicate DEPOLYMERIZE. In addition, to be able to determine the effect attribute in the binarisation, relationship types considered regulatory (Dynamics and Amount subtrees and the PREVENT type) were flagged.

4.2 Statistics

This section briefly summarises the key statistics relating to the binarisation. The original BioInfer corpus in the graph representation contains 2662 complete relationships, 942 of which are binary. Note that some of these binary relationships (such as EQUAL) are preprocessed into other relationships. The binarised BioInfer contains 2762 relationships of which 94.4% (vs. 93.9% in the original) have positive polarity and 89.7% direct action effect.

During the binarisation process, the rules matched 4794 times in total: the fraction of rules involving the root is 39.7% and those involving leaves 51.6%. The most applied root-matching rules were those processing CAUSE, regulatory relationship types, and CONTAIN (10.3%, 9.8%, 8.4% resp.) while leaf-matching rules were applied mostly to remove edges of identity (21.3%) or structure/membership (17.3%) types.

The distributions of predicates in the original and binarised BioInfer are clearly different. In the binarised corpus, general predicates (for example PARTICIPATE, AFFECT, and CONDITION) have nearly all been removed while the number of predicates in the Change-subtree has increased 63% even though the number of predicates in its Dynamics-subtree have decreased 25%. The former two observations confirm that the general predicates have been transformed to biologically relevant ones, as intended. The last observation corresponds to the regulatory predicates being reinterpreted as effect attributes.

4.3 Error analysis

Table 2 shows the observed errors and approximations in the sample. For those types that can occur only once per relationship, the expected number per relationship in the binarised BioInfer is shown. For the lost interaction details, the expected number per non-leaf entity in the original BioInfer is shown.

Three of the observed missing interactions are

error type	count	E
missing interaction	7	0.07
invalid inference	13	0.12
invalid interaction text binding	0	0.00
total	20	0.19

approximation type	count	E
approximated interaction type	8	0.08
lost entity (process/property)	9	0.06
lost entity (structure/membership)	15	0.09
lost entity (identity)	7	0.04
total	31	0.19

Table 2: The errors and approximations observed in the analysed sample of the binarised BioInfer. Expectation E for errors and approximated interaction types given per-relationship, other approximations per-relation, where per-relationship expectations refer to the binarised corpus and per-entity expectations to the original corpus.

duplicates of existing interactions. For example, two regulatory relationships would be annotated in the sentence *Actin regulates cofilin phosphorylation and dephoshorylation*, but the binary annotation cannot express the difference and hence produces only one relationship. Another three missing interactions are deliberately removed as self-interactions (which are not relationships in the applied semantic network model). The last missing interaction is due to the failure in nesting role resolution, caused by an invalid nesting in a phrase *actin-bound nucleotide exchange*. The nesting is technically allowed by the BioInfer annotation but the role of *actin* in *exchange* cannot be expressed with a single edge.

For the majority of the observed invalid inferences the cause is an incorrectly identified effect attribute. In six cases, the regulatory effect of a node is missed or falsely assumed. For example, in the sentence *Addition of profilin caused actin depolymerisation*, the process *addition* (annotated as INCREASE) does not refer to positive regulation but rather to an experimental setup. The two other effects are misidentified due to a similar case of nesting as described in the previous paragraph (consider the phrase *concentration required for polymerisation*). In the remaining five cases, the true agent is an unexpressed process while the claimed agent (protein) has an unstated relationship with the patient. This renders the binarised relationship invalid. Consider the sentence *Dephosphorylation of cofilin leads to actin depolymerisation* as an example in which *dephosphorylation* causes *depolymerisation* while the effect of *cofilin* as such on *actin* is unstated.

The expectations for losing information in entities is surprisingly low given that leaf-targeting rules were the most applied. Moreover, since words carrying little biologically relevant information, such as "protein" and "function", are included in these numbers, the biological information loss is even less. The observed approximations in the interaction types are minor, such as the type INITIATE being generalised to positive regulation in mapping to an attribute.

In short, the error analysis reveals some weaknesses of the original BioInfer annotation scheme, especially nesting, while the binarisation fails mostly on identifying a regulatory effect. Given that regulatory relationships are a small minority, the effect attribute could be completely dismissed.

5 Conclusions

This paper has provided the first study of the relationship between the pairwise annotations commonly used to annotate PPI and the complex annotations in recent corpora such as BioInfer and the GENIA Event corpus. A simple semantic network representation was presented, and the BioInfer predicate annotation was mapped into this representation. This mapping unifies some arguably unnecessary distinctions in the original annotation, such as the mirroring of some relationship types with entity types (e.g. PHOSPRORYLATE vs. PHOSPHORYLATION), and explicitly represents all relationships between entities, including relationships whose type is unspecified in the original annotation (e.g. sub/superstructure). The semantic network thus provides a more consistent representation of the relevant information, facilitating rule-based inference.

The binarisation of the BioInfer relationship annotation was implemented as a set of graph transformation rules. This transformation aimed to determine which biologically relevant relationships between two proteins can be inferred from the full semantic network and how much of the original information content can be preserved with BioInfer relationship types augmented with polarity and effect (direct/regulatory) attributes. A study of the resulting binary PPI indicated that

while the original annotation and the chosen representation are, in general, capable of supporting this form of inference, a number of errors were produced in the process. The study of these errors suggested some weaknesses in the original annotation and further indicated that while the existence of relationships was inferred correctly, the effect attribute could not always be reliably determined. The evaluation further provided an estimate of the approximations inherent to binary annotation even when regulatory effects are separately captured.

The results suggest that it is sufficient to summarise the relationships between proteins with a pairwise annotation for use in various applications. However, information extraction could benefit from the details available in complex relationships. Thus, together with the possibility to transform complex relationship into binary ones, the extraction of semantic networks could prove to be a feasible approach to PPI information extraction.

The similarities between the network representation considered here and the conceptual graph (CG) model of Sowa (1976) suggest that the CG model could be adopted as a knowledge representation for PPI extraction. As a well-founded formalism, the CG model would provide a means to robustly express extracted relationships. However, the CG model may need to be adjusted to address the linguistic aspects of information extraction in the biomedical domain.

The created binary BioInfer is the first corpus with pairwise PPI annotation where the rationale for including or excluding a particular pair is formalised to the level of computationally implemented rules. As binary PPI annotation is still dominant in particular in machine-learning-based PPI extraction, this resource can provide valuable data to a field where annotation consistency has been a challenge. Similarly, the semantic network form of the corpus can provide a more approachable target for automatic PPI extraction than the original predicate form. The software tools and the data (in the original BioInfer format) produced in this study are freely available from `http://www.it.utu.fi/BioInfer`.

Acknowledgements

This work has been supported by the Academy of Finland.

References

Bea Alex, Claire Grover, Barry Haddow, Mijail Kabadjov, Ewan Klein, Michael Matthews, Stuart Roebuck, Richard Tobin, and Xinglong Wang. 2008. The ITI TXM corpora: Tissue expressions and protein-protein interactions. In *Proceedings of LREC'08*.

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

J. Ding, D. Berleant, D. Nettleton, and E. Wurtele. 2002. Mining MEDLINE: abstracts, sentences, or phrases? In *Proceedings of PSB'02*, pages 326–337.

Carol Friedman, Pauline Kra, Hong Yu, Michael Krauthammer, and Andrey Rzhetsky. 2001. GENIES: A natural-language processing system for the extraction of molecular pathways from journal articles. *Bioinformatics*, 17(Suppl. 1):S74–S82.

Jin-Dong Kim, Tomoko Ohta, and Jun'ichi Tsujii. 2008. Corpus annotation for mining biomedical events from literature. *BMC Bioinformatics*, 9(10).

Martin Krallinger, Florian Leitner, and Alfonso Valencia. 2007. Assessment of the second BioCreative PPI task: Automatic extraction of protein-protein interactions. In *Proceedings of BioCreative II*, pages 41–54.

Ryan McDonald, Fernando Pereira, Seth Kulick, Scott Winters, Yang Jin, and Pete White. 2005. Simple algorithms for complex relation extraction with applications to biomedical IE. In *Proceedings of ACL'05*, pages 491–498.

Igor A. Mel'čuk. 1988. *Dependency Syntax: Theory and Practice*. State University of New York Press.

Claire Nédellec. 2005. Learning language in logic - genic interaction extraction challenge. In *Proceedings of LLL'05*.

Sampo Pyysalo, Filip Ginter, Juho Heimonen, Jari Björne, Jorma Boberg, Jouni Järvinen, and Tapio Salakoski. 2007. BioInfer: A corpus for information extraction in the biomedical domain. *BMC Bioinformatics*, 8(50).

Sampo Pyysalo, Antti Airola, Juho Heimonen, Jari Björne, Filip Ginter, and Tapio Salakoski. 2008. Comparative analysis of five protein-protein interaction corpora. *BMC Bioinformatics*, 9(Suppl. 3):S6.

John F. Sowa. 1976. Conceptual graphs for a database interface. *IBM Journal of Research and Development*, 20(4):336–357.

From Terms to Categories: Testing the Significance of Co-occurrences between Ontological Categories

Robert Hoehndorf
Institute for Medical Informatics, Statistics and Epidemiology and
Department of Computer Science, University of Leipzig and
Department of Evolutionary Genetics, Max-Planck-Institute for Evolutionary Anthropology
hoehndorf@eva.mpg.de

Axel-Cyrille Ngonga Ngomo
Department of Computer Science, University of Leipzig
ngonga@informatik.uni-leipzig.de

Michael Dannemann and **Janet Kelso**
Department of Evolutionary Genetics
Max-Planck-Institute for Evolutionary Anthropology
{michael_dannemann,kelso}@eva.mpg.de

Abstract

The co-occurrence of terms in a text corpus may indicate the presence of a relation between the referents of these terms. We expect co-occurrence-based methods to identify association relations that cannot be found using static patterns. We developed a new method to identify associations between ontological categories in text using the co-occurrence of terms that designate these categories. We use the taxonomic structure of the ontologies to cumulate the number of co-occurrences of terms designating categories. Based on these cumulated values, we designed a novel family of statistical tests to identify associated categories. These tests take both co-occurrence specificity and relevance into consideration. We applied our method to a 2.2 GB text corpus containing fulltext articles and used Gene Ontology's biological process ontology and the Celltype Ontology. The software and results can be found at http://bioonto.de/pmwiki.php/Main/ExtractingBiologicalRelations.

1 Introduction

An increasing number of biomedical ontologies address the problem of biological data integration. Ontologies are a means for organizing and representing basic categories and relationships pertaining to the conceptualization of a domain. Many biomedical ontologies have been developed according to a common set of criteria based on the Open Biomedical Ontologies (OBO) or the OBO Foundry. A common property of these ontologies is their focus on a single domain. This particular property provides an easy means for applying an ontology to a domain-specific application. However, knowledge bridging multiple domains remains hidden and not explicit.

To address this problem, so-called "cross-products" have been created. They define categories from one ontology using categories from other ontologies and relations from the OBO Relationship Ontology (RO) (Smith et al., 2005). Due to the large number of categories in the OBO ontologies, few of these cross-products exist and are maintained. For example, parts of the Gene Ontology (GO) (Ashburner et al., 2000) are defined using categories of cells from the Celltype ontology (CL) (Bard et al., 2005) and relations like **has-participant** from the RO. While many of these cross-products have been created in a manual curation effort, some were created using automated information extraction methods (Bada and Hunter, 2007), which exploit the compositional nature of many terms in these ontologies.

Methods based on term decomposition can provide high quality logical definitions suitable for inclusion in a stable version of the ontology. Yet, they miss several more intricate relations between categories that are not reflected in their names. For example, the relation between *cardiac muscle cells* (CL:0000746) and *heart looping* (GO:0001947) cannot be uncovered using basic pattern matching. Other approaches have been

used to extract relations between categories in ontologies. Among them are association rule mining and statistical analysis of term co-occurrences (Bodenreider et al., 2005).

In this paper, we present a novel method for extracting association relations between categories defined in distinct biomedical ontologies. This method takes as input a set of ontologies and a text corpus. It then detects associations between categories of the input ontologies based on the co-occurrence of terms that designate ontological categories. The data obtained by analyzing co-occurrences is further refined according to the structure of the input ontologies. The resulting association relations can either be considered by human curators, used as input for automated relationship extraction methods or exploited by question answering systems.

2 System and Methods

2.1 Ontologies

An ontology is the specification of a conceptualization of a domain (Herre et al., 2006). Many biological ontologies are represented as directed acyclic graphs (DAGs) and are available in the OBO flatfile format [1]. In these DAGs, nodes represent *categories* and edges represent *relations* between these categories. A category, also called *kind*, *class* or *universal*, is an entity that is general in reality. Examples are *dog*, *apoptosis*) or *to transport sugar*. Categories may have instances, of which some may not be further instantiated. These are called *individuals*. We call the set of all categories in an ontology O $Cat(O)$.

Categories may be related to other categories. The most important relation between two categories A and B is the **is-a** relation, $isA(A,B)$. The relation $isA(A,B)$ can be defined using the instantiation relation: when $isA(A,B)$, then all instances a of A are instances of B (Herre et al., 2006). This definition implies that the **is-a** relation is reflexive and transitive.

A set of categories with the **is-a** relation among them form a taxonomy. These taxonomies often are the backbone of the OBO ontologies' DAG structure. We call the set of all successors of a category A the sub-categories $subcat(A) = \{B|isA(B,A)\}$ and its predecessors the super-categories $supcat(A) = \{B|isA(A,B)\}$. The direct successors and predecessors of A in the taxonomy are called children ($child(A) = \{B|isA(B,A) \wedge B \neq A \wedge \forall X(isA(B,X) \wedge isA(X,A) \rightarrow X = B)\}$) and parents, respectively.

In the OBO flatfile format, ontologies are assigned a namespace. Category-identifiers are prefixed with the namespace of the ontology to which they belong. Therefore, they are unique within the OBO ontologies. In addition to a unique identifier, categories are assigned a *name* and a set of *synonyms*. Neither the name nor the set of synonyms must be unique.

2.2 Basic Assumptions

Our method for extracting association relations between categories is based on two main assumptions:

1. Terms can designate ontological categories; the terms that designate the same category are henceforth called the category's synset. Every occurrence of an element of the synset of category C is called an occurrence of C. Every co-occurrence of an element of the synset of the category C with an element of the synset of the category D is called a co-occurrence of C and D.

2. When A is a sub-category of B, then every co-occurrence of A with C is a co-occurrence of B with C. Additionally, every occurrence of A counts as an occurrence of B.

According to our first assumption, we constructed synsets from the synonyms attached to each category in the input ontologies, and counted the occurrences and co-occurrences of these synsets based on two contexts: single sentences and sentences in documents[2]. We used exact matching to identify terms in text. Secondly, we computed the closure of the occurrences and co-occurrences of the categories with respect to the **is-a** relation, as explicated in our second assumption.

Finally, we test for the collocation between categories based on the occurrence and co-occurrence of elements of their synsets. Here, collocation refers to a co-occurrence that is higher

[1] http://www.cs.man.ac.uk/~horrocks/obo/

[2] The second context refers to whole documents, but co-occurrence is based on single sentences. Therefore, when two terms co-occur in two or more sentences within one document, their co-occurrence is only counted once.

than expected by chance. To this end, we designed a family of tests that account for both the ontologies' structure and the term distribution in the text corpus. The tests account for both relevance and specificity of the co-occurrence of categories. In this context, relevance refers to how often the categories co-occur in the text corpus compared to their absolute occurrence. The second aspect of the tests allows the identification of the categories that contain the most information within the ontologies, i.e., they are the most specific categories with respect to the **is-a** relation.

To test our method, we used the biological process (BP) branch of the Gene Ontology (GO) (Ashburner et al., 2000) and the Celltype Ontology (CL) (Bard et al., 2005). Our experiments were conducted using a 2.2 GB text corpus containing 60143 fulltext articles from Open Access journals listed in Pubmed Central.

2.3 Method

We first analyzed the text corpus for the occurrence and co-occurrence of the terms included in the synsets of categories taken from two ontologies. Based on these values, we computed the occurrence and co-occurrence values for the categories. To test the statistical significance of these co-occurrence values, we generated several permutations of the data extracted from the text corpus. These approximate a random distribution of co-occurrence values within the ontologies for the chosen text corpus. We then calculated the p-values for the observed values against this random distribution. Finally, we applied a family of novel tests to these p-values to identify collocated categories from the ontologies. The result of our approach is a list containing pairs of categories that are collocated with respect to a given cutoff.

2.3.1 Text Processing

First, we counted the number of occurrences and co-occurrences of the terms contained in synsets of categories from the input ontologies. We counted the total number of sentences and documents in which at least one element of a synset was found using exact matching. For each pair of categories, we counted the total number of co-occurrences of elements of their respective synsets in sentences. Furthermore, we counted the number of documents in which they co-occured within at least one sentence. We used exact matching and abstained from using any more sophisticated methods for recognizing the ontologies' categories in text at this point in time.

The text processing yielded, for each category C, both its frequency $f(C)$ (total number of occurrence of terms from $syn(C)$ in sentences) and the total number of documents in which an element from $syn(C)$ appeared, $d(C)$. Furthermore, for each pair of categories C_1 and C_2, we obtained both the total number of co-occurrences in sentences $f(C_1,C_2)$ and the total number of documents containing these co-occurrences $d(C_1,C_2)$.

2.3.2 Co-occurrence Cumulation Using Ontologies

The second step in our method implemented our second assumption, i.e., occurrence and co-occurrence between categories is transitive over the **is-a** relation. We assumed that when two categories C and C' stand in the **is-a** relation, C **is-a** C', then every occurrence of C is also an occurrence of C'. This means that the synset-closure $synclos(C)$ of a category C can be constructed as follows:

$$syn(C) \subseteq synclos(C) \quad (1)$$

$$isA(C,C') \rightarrow (syn(C) \subseteq synclos(C')) \quad (2)$$

For all categories C, the values $f_t(C)$ and $d_t(C)$ represent the sum of the values $f(C')$ and $d(C')$ over all of C's sub-categories C'. For all categories C_1 and C_2, we computed the cumulated f- and d-values dubbed $f_t(C_1,C_2)$ and $d_t(C_1,C_2)$:

$$f_t(C_1,C_2) := \sum_{a \in subcat(C_1)} \sum_{b \in subcat(C_2)} f(a,b), \quad (3)$$

$$d_t(C_1,C_2) := \sum_{a \in subcat(C_1)} \sum_{b \in subcat(C_2)} d(a,b), \quad (4)$$

For all categories C_1 and C_2, we defined the following score function:

$$score(C_1,C_2) = \frac{\log f_t(C_1,C_2)}{\log(1+f_t(C_1)) + \log(1+f_t(C_2))} \cdot \frac{\log(d_t(C_1,C_2))}{\log(1+\max(d_t(C_1),d_t(C_2)))} \quad (5)$$

The first component of the score function implements the natural logarithm of the Pointwise Mutual Information (PMI) (Manning and Schütze, 1999) score achieved by the categories with respect to their co-occurrence within sentences. In order to avoid divisions by 0, the denominators

of all members of the score function were incremented. The second component measures a similar value using documents as context. The aim of the score function is to ensure that categories that co-occur relatively often are assigned a high score. The range of the score function is between 0 and 1, and categories with overlapping synsets will have a score of 1.

2.3.3 Determining the Random Distribution

The score of two categories C and D is influenced by the topology of the ontology: categories that are more general occur and co-occur more often, due to our definition of occurrence and co-occurrence of categories. Therefore, it is insufficient to test for a high score to consider the co-occurrence of two categories as significant. A random distribution for the scores of each pair of categories C and D provides a means for determining the significance of a co-occurrence. This random distribution depends on the text corpus, the method for identifying categories, the score function and the topology of the ontologies. Hence, we did not assume any statistical distribution of scores.

We simulate the random distribution of the scores of each category pair through multiple random permutations: the f- and d-values that were measured for each synset during the first step of our method were randomly assigned to categories in the ontology from which they originated. We then calculated and recorded co-occurrence scores for all pairs of categories. In addition, for each category D, such that $isA(D,C_1)$, the score difference $score(C_1,C_2) - score(D,C_2)$ was recorded. Further, for each category E with $isA(C_1,E)$, the score difference $score(E,C_2) - score(C_1,C_2)$ was recorded.

Hence, the results of this step were threefold. First, we approximated the random score distribution for each pair of categories. Second, each triple of categories C, D and $E \in child(C)$ gave rise to a random distribution of score differences between (C,D) and (E,D). Third, each triple C, D and $E \in parent(C)$ yielded a random distribution of score differences between (E,D) and (C,D).

2.3.4 Significance Testing

To identify strong co-occurrences, we designed a family of tests for each co-occurrence that considers a fragment of the path in the ontology graph. The first kind of tests is assymetrical. At the end of this section, we will introduce a symmetrical form of these tests. The first tests are designed to test the significance of the co-occurrence between C_1 and C_2 based on three criteria: (1) the score $score(C_1,C_2)$ for the co-occurrence should be higher than expected; (2) for each child category D of C_1, $score(C_1,C_2) - score(D,C_2)$ should be higher than expected and (3) for each parent category E of C_1, $score(E,C_2) - score(C_1,C_2)$ should be lower than expected.

The first criterion measures relevance, while criteria (2) and (3) test for specificity. The first criterion establishes high confidence in the co-occurrence strength. The second criterion reflects the assumption that a collocation must be novel, i.e., it must represent an information increase over the co-occurrences of a sub-category. Therefore, given that $isA(D,C_1)$, we assume that any relevant information obtained from the co-occurrence between C_1 and C_2 already appears in the co-occurrence between D and C_2 when the difference between $score(C_1,C_2)$ and $score(D,C_2)$ is low (with respect to the random distribution of scores). We would assume a collocation between D and C_2, because D is more specific than C_1. On the other hand, if the difference between $score(C_1,C_2)$ and $score(E,C_2)$ is high (with respect to the random distribution of scores), and $isA(C_1,E)$, we would assume a collocation between E and C_2. We describe the intuitions behind our tests below. The complete description and formalization of the tests can be found on the project website.

Within this section, let C and D be fixed categories from ontologies O_1 and O_2, respectively. Furthermore, let N be the number of permutations.

The first test we designed depends on the categories C and D, the ontology's structure and the number of permutations N. It tests for the following properties:

- the co-occurrence score between C and D is high,
- the difference between $score(C,D)$ and $score(C',D)$ for every child C' of C is high,
- the difference between $score(C,D)$ and $score(C'',D)$ for every parent C'' of C is low.

"Being high" and "being low" were captured using the values of the cumulative distribution functions (CDFs) obtained by the N permutations performed in the previous step: one function for each pair of categories C and D, one function for each triple of categories C, D and C' where C' is a child of C, and one for each triple C, D and C'' where C'' is a parent of C. We then combined the p-values of the score differences to children in a single value using their geometric mean. A similar combination of the score differences' p-values to the parent categories of C was carried out: here, the combined value is the geometric mean of $1-x$, where x is the p-value in the corresponding CDF.

The geometric mean was used because it has properties that correspond to our intuitions: when the score difference to one of the child categories is very low (the p-value in the CDF is 0), we always prefer the co-occurrence of the child of C and D over the co-occurrence of C and D. The geometric mean would then be 0, and the result of the first test Θ_1 would be 0 as well. Very high differences (the p-value in the CDF is 1) are ignored, i.e., the value of the geometric mean depends solely on the other child categories of C.

The inverse holds for the score differences between C and D and the parents of C and D: when the p-value of the score difference in the CDF is 0, this difference is ignored (because $1-0=1$, and thus does not heavily influence the value of the geometric mean), while a high difference (the p-value in the CDF is 1) results in a final score of 0.

The goal of the test Θ_1 is to find the *most specific* pair of categories that co-occur significantly often. Therefore, the score between the two categories should be high, and provide a significant increase over all the child categories. If there was no such increase, i.e., the score between C and D is high and the score between the children of C and D is high as well, Θ_1 prefers the co-occurrences between children of C and D, because they are more specific and therefore contain more information. The difference to the parents of C should be low, as otherwise there would be a significant increase in the score between a parent of C and D over the score between C and D. Then, Θ_1 prefers the co-occurrence between this parent and D over C and D.

All other tests are extensions of the first test. The second test, Θ_2, uses the minimum function instead of the geometric mean to combine the p-values in the CDFs of the score differences to parents and children.

The first two tests Θ^1 and Θ^2 do not consider the variances of the distributions of scores, differences in scores to children and differences in scores to parents. Therefore, we extended these tests by weighting all three components of the tests with the variances of their corresponding distributions. In these tests, high variance lowers the impact of the result, while lower variance strengthens it.

We defined three new distributions for the variances, and chose the p-value in the respective CDF as a weight in our tests. We computed the scores for each pair of category N times, resulting in one distribution of scores for each pair of categories. Each of these distributions has a variance. The score variance distribution is the finite distribution (containing N elements) of the variances of each of these distributions. We defined the variance distribution for score difference to parent and child analogously.

The tests Θ^3 and Θ^4 use only the variance distribution of scores, while Θ^5 and Θ^6 use all three variance distributions. These tests are one-sided, i.e., they are not symmetric. We define two-sided, symmetric tests $\tau^i(C,D)$ for all categories C and D as

$$\tau^i(C,D) = \Theta^i(C,D) \cdot \Theta^i(D,C) \qquad (6)$$

3 Implementation

The text processing module is implemented in Java. The remaining steps are implemented using a combination of Java classes and Groovy scripts. The source code for all programs is available under the modified BSD license from the project webpage. The implementation uses the functionality of the GNU GetOpt library[3], Java Universal Network/Graph Framework[4] and the Java Colt libraries[5].

4 Discussion

4.1 Results

We applied the method described here to the biological process (BP) branch of the Gene Ontology

[3] http://sourceforge.net/projects/evcgen/

[4] http://jung.sourceforge.net

[5] http://dsd.lbl.gov/~hoschek/colt/

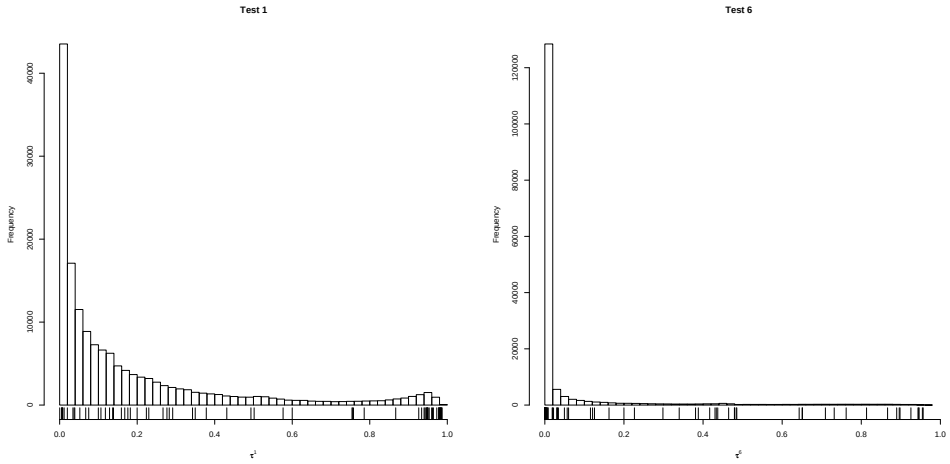


Figure 1: Distribution of test results. The plot on the left shows the distribution of the test results for τ^1. On the right, the same is shown for τ^6. The tests using the minimum function ($\tau^{2,4,6}$) are stronger than the tests using the geometric mean ($\tau^{1,3,5}$). Furthermore, weighting the tests with the CDFs of the variances ($\tau^{3,4,5,6}$) produces stronger results than the basic tests ($\tau^{1,2}$). Below the distributions, the quantiles of the GO-CL dataset for each test are displayed.

p	τ^1	τ^2	τ^3	τ^4	τ^5	τ^6
0.5	0.075	0.017	0.024	0.003	0.007	0.001
0.8	0.288	0.145	0.141	0.047	0.061	0.016
0.9	0.522	0.433	0.298	0.168	0.220	0.120
0.95	0.806	0.790	0.472	0.412	0.456	0.400
0.99	0.952	0.950	0.863	0.826	0.859	0.824

Table 1: The table shows p-quantiles for different p-values for all six tests. Given a p-value (first column), the quantiles show the result of each test for which p-values are below the quantile.

(GO) and the Celltype Ontology (CL). We identified 3,751 out of the 14,542 terms in the GO's biological process ontology in our text corpus. We found 491 of 754 terms from the CL. Terms from the GO's BP branch co-occurred 70,967 times with CL terms.

Using our method, we identified a total number of 202,627 co-occurrences between categories. After applying our tests, 157,894 co-occurrences produced p-values distinct from 0.[6] We illustrate the quantiles obtained for different p-values in our six tests, τ^i, in table 1. The distribution of scores for τ^1 and τ^6 are shown in figure 1. The remaining plots are available on the project webpage.

We found that the tests using the minimum instead of the geometric mean of p-values of score differences to parent and child categories are generally stronger, i.e., they include fewer co-occurrences as significant for a given cutoff. Similarly, tests including the variance for scores are generally stronger than tests that are not weighted by the variance of score distributions. In this sense, the tests τ^5 and τ^6 are the strongest.

Relation	Number of occurrences
has-participant	62
Participates-in	13
Located-in	2
unclassified	38

Table 2: Manually identified ontological relations in the 100 top-scoring association results (with respect to τ^1).

Table 2 shows the kind of relationship between categories that our tests identified for the 100 top-scoring results with respect to test τ^1. The *has-participant* relation is defined in (Smith et al., 2005). We define the *Participates-in* relation as: C_1 *Participates-in* $C_2 \iff \forall x,t_1(instanceOf(x,C_1,t_1) \rightarrow \exists t_2,y(instanceOf(y,C_2,t_2) \wedge participates\text{-}in(x,y,t_2)))$, where *participates-in* is the primitive participation relation between individuals as defined in (Smith et al., 2005). We extend the definition of *located-in* in (Smith et al.,

[6]The remainder obtained a score of 0 due only to numerical restrictions. They were subsequently excluded, because they were indistinguishable from the absence of co-occurrence.

2005) to a relation *Located-in* between processes and objects, which holds when all participants of a process are *located-in* a structure during the entire duration of the process.

In our sample, 38 association relations do not fall under one of the three relations that we investigated. We discovered several kinds of unclassified relations. First, mismatches in granularity lead to strong associations for unrelated categories. For example, *xanthine transport* and *erythrocyte* are closely related according to τ^1. Erythrocytes are involved in the transport of xanthine. However, the GO category *xanthine transport* refers to the inter- and intracellular level of granularity, while erythrocytes transport nutrients between organs. Second, some categories are indirectly related via another category. For example, osteoclasts and lymph node development are related via the protein RANK. Third, when cells have closely related functions, we identify too specific or too generic cell types as in the case of the association between *basophil degranulation* and *mast cell*. Finally, 6 out of 100 associations in our sample seem erroneous.

4.2 Comparison with Other Approaches

We did not compute precision or recall for our method, due to the absence of a gold standard. However, we compared our method with the GO-CL crossproducts available[7] from the OBO Foundry[8]. The dataset contains manually verified relations between categories from the GO and the CL that have been extracted using the method described in (Bada and Hunter, 2007). Because this method is based on the compositional nature of terms in the GO, it exclusively identifies relations in which one category name (usually a type of cell) is a substring of another category name (usually a GO category).

The GO-CL crossproduct contains 396 relations between GO and CL categories. From these 396, we identified 73 that co-occurred in our text corpus. Table 3 shows the percentage of significant co-occurrences within these 73 relations for different cutoffs in our six tests. Figure 1 shows the distribution of the 73 pairs with respect to τ^1 and τ^6.

As our method relies exclusively on the distribution of terms and not on their syntactic structure, it permits the recognition of association relations between categories that could not be recognized using patterns. An example of such an association is *myoepithelial cell* (cells located in the mammary gland) and *milk ejection*.

However, while (Bada and Hunter, 2007) identified well-defined, ontological relations, our approach is designed to identify strongly associated categories that can be further refined using complementary approaches for identifying relationships from text, such as abductive reasoning (Hobbs et al., 1988).

Recall	τ^1	τ^2	τ^3	τ^4	τ^5	τ^6
95%	0.007	0.006	0.003	0	0.002	0
80%	0.102	0.054	0.028	0.003	0.016	0.002
70%	0.173	0.109	0.049	0.008	0.029	0.004
50%	0.502	0.350	0.173	0.063	0.154	0.060

Table 3: Evaluation of our approach with respect to the GO-CL dataset (Bada and Hunter, 2007). The dataset we used for comparison consists of the 73 relations from (Bada and Hunter, 2007) found in our text corpus. Columns two to seven show the cutoff values required to identify the percentage given in column one of relations as significant using tests one to six.

4.3 Future Research

The method presented in this paper can be enhanced by several means. First, our term identification approach could be improved. A large number of variants of the terms included in the synset of each category may occur in scientific texts. Since our term recognition is based on exact matching, we expect to miss a large number of term occurrences and other references to the ontologies' categories. In particular, this affects the recognition of terms from the GO. We expect that the integration of methods such as (Gaudan et al., 2008) for recognizing GO categories in text would improve our results. Further natural language processing techniques such as stemming could improve the identification of categories in text.

Second, we currently estimate the random score distribution throughout the ontologies using multiple permutations. A deeper statistical analysis could provide insights on how to replace the random distributions obtained through permuta-

[7] `http://obofoundry.org/cgi-bin/detail.cgi?id=go_xp_cell`, accessed on January 23rd, 2008.

[8] `http://obofoundry.org`

tions with the exact random distributions. We expect this to improve the accuracy of our method.

The main goal of our future research will be to extract well-defined, ontological relations between categories. The method we propose in this paper serves as the first step in such an effort, because it generates relevant associations according to the scientific literature used. Additional methods that may be based on the manual generation of patterns (Bada and Hunter, 2007), pattern learning (Hao et al., 2005) or the application of methods from logics and ontologies (Schulz et al., 2006; Hobbs et al., 1988) could then be applied.

In the meantime, we plan to apply our method to other ontologies and lexical resources.This is possible because our method uses directed graph structures, in which edges represent relations from a less specific to a more specific entity. Such a graph structure can be extracted from a wide variety of biomedical resources.

4.4 Conclusion

We developed a novel method to identify association relations between ontological categories from co-occurrences between terms obtained using text-mining techniques. For this purpose, we have implemented a suite of tools that can be used to extract these association relations from a text corpus and two ontologies represented in the OBO flatfile format. To evaluate the strength of the association relations between the ontological categories, we designed a family of novel statistical tests that account for the ontologies' topologies and test for relevance and specificity.

We applied our method to extract several thousands of associated categories from the Gene Ontology and the Celltype Ontology using a text corpus comprised of fulltext scientific articles from PubMed Central. The association relations that we extracted are available for downloadat `http://bioonto.de/pmwiki.php/Main/ExtractingBiologicalRelations`.

Acknowledgement

We thank Leonardo Bubach, Hernán Burbano and Heinrich Herre for helpful discussions and valuable comments, and Christine Green for her help in preparing the manuscript.

References

M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. 2000. Gene ontology: tool for the unification of biology. the gene ontology consortium. *Nat Genet*, 25(1):25–29, May.

M. Bada and L. Hunter. 2007. Enrichment of obo ontologies. *Journal of Biomedical Informatics*, 40(3):300–315, June.

J. Bard, S. Y. Rhee, and M. Ashburner. 2005. An ontology for cell types. *Genome Biology*, 6(2):R21.

O. Bodenreider, M. Aubry, and A. Burgun. 2005. Non-lexical approaches to identifying associative relations in the gene ontology. *Pac Symp Biocomput*, pages 91–102.

S. Gaudan, A. Jimeno Yepes, V. Lee, and D. Rebholz-Schuhmann. 2008. Combining evidence, specificity, and proximity towards the normalization of gene ontology terms in text. *EURASIP Journal on Bioinformatics and Systems Biology*, 2008(3):9.

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

H. Herre, B. Heller, P. Burek, R. Hoehndorf, F. Loebe, and H. Michalek. 2006. General Formal Ontology (GFO) – A foundational ontology integrating objects and processes [Version 1.0]. Onto-Med Report 8, Research Group Ontologies in Medicine, Institute of Medical Informatics, Statistics and Epidemiology, University of Leipzig, Leipzig, Germany.

J. R. Hobbs, M. Stickel, P. Martin, and D. D. Edwards. 1988. Interpretation as abduction. In *26th Annual Meeting of the Association for Computational Linguistics: Proceedings of the Conference*, pages 95–103, Buffalo, New York.

Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts.

S. Schulz, E. Beisswanger, J. Wermter, and U. Hahn. 2006. Towards an upper-level ontology for molecular biology. *AMIA Annu Symp Proc*, 2006:694–698.

B. Smith, W. Ceusters, B. Klagges, J. Köhler, A. Kumar, J. Lomax, C. Mungall, F. Neuhaus, A. L. Rector, and C. Rosse. 2005. Relations in biomedical ontologies. *Genome Biol*, 6(5).

Towards Automatic Detection of Experimental Methods from Biomedical Literature

Thomas Kappeler, Simon Clematide, Kaarel Kaljurand, Gerold Schneider, Fabio Rinaldi*
Institute of Computational Linguistics, University of Zurich, Switzerland
kappeler@bluewin.ch,
{siclemat, kalju, gschneid, rinaldi}@ifi.uzh.ch

Abstract

In this paper we present techniques aimed at detecting, within scientific papers which describe newly discovered protein interactions, the methods used by the authors of the research to experimentally verify the interaction(s).

We compare previous results over the BioCreAtIvE data set with more recent results over a larger data set, using INTACT annotations as gold standard. This comparison shows the generality of the proposed approach and suggests that practical application of these techniques within a curation environment might not be that far away.

1 Introduction

Protein interactions play fundamental roles in biological processes (e.g. signal transduction). Biologists routinely perform experiments in order to detect or confirm protein interactions. In doing so, they use a variety of experimental methods.

Databases such as INTACT (Kerrien et al., 2006) or MINT (Zanzoni et al., 2002) aim at collecting the known interactions from the literature. The process of extracting selected items of information from the published literature in order to store such items in databases is known as "curation". This is a costly and time-consuming process, which still requires a significant amount of human resources to be performed effectively.

Tools that can support the process of curation would be extremely welcome by the community. Such tools should be capable of detecting within the papers, with high reliability, all the information that the curators need to create database records.

Repositories of protein interactions, such as INTACT and MINT, store, together with each interaction, a reference to the experimental method that was used to detect it, because this information is highly relevant to researchers. Therefore, not only the protein interactions, but also the experimental methods, need to be identified.

There is a limited number of available experimental methods for the detection of protein interactions, which are all described within the PSI-MI taxonomy (Hermjakob et al., 2004), in particular under node MI:0001 (interaction detection method). Each method is provided with a unique numerical identifier, a standard name, a definition and a list of synonyms.[1]

In order to stimulate research aimed at developing tools that support the extraction of critical information from the literature, the recent BioCreAtIvE text mining competition set up a number of tasks which partially simulate the process of curation. In particular the Protein-Protein Interaction task (PPI) was organized in four subtasks (Krallinger et al., 2008): PPI-IAS (identification of abstracts which contains curatable protein-protein interactions), PPI-IPS (identification of protein-protein interactions in abstracts), PPI-ISS (identification of sentences which provide evidence for protein-protein interactions), PPI-IMS (identification of the experimental method by means of which the interaction was verified). Our own participation to BioCreAtIvE focused on the IPS and IMS subtasks (Rinaldi et al., 2008).

In this paper we describe recent experiments aimed at testing the coverage of the IMS detection approach across a larger set of articles.

*Corresponding author

[1] For the experiments described in this paper we used version 2.5 of PSI-MI.

MI:0096 (pull down)	20.6%
MI:0007 (anti tag coip)	13.1%
MI:0018 (two hybrid)	12.7%
MI:0006 (anti bait coip)	12.1%
MI:0019 (coip)	8.8%
total	67.3 %

Table 1: The 'Big5': most frequently occurring methods in the BioCreAtIvE training data

Run	R	P	F
run 1	29.4%	65.4%	40.6%
run 2	56.8%	43.5%	49.3%
run 3	53.9%	51.3%	52.6%
run 1	20.02%	66.79%	30.81%
run 2	43.02%	40.34%	41.64%
run 3	40.96%	49.65%	44.89%

Table 2: Above: our best results over BioCreAtIvE training data. Below: our official results over BioCreAtIvE test data[3]

2 Detection of Experimental Methods

In the original BioCreAtIvE setting, the organizers asked the participants to deliver the methods coupled with the interactions to which they apply. Due to the intrinsic difficulty of the problem, coupled with the difficulty of finding the interactions, the task was later relaxed, and the participants were asked to deliver a set of experimental methods employed in the article.

The approach we used in BioCreAtIvE for the detection of the experimental methods is based on pattern matching supplemented by simple statistics. As it would have been impossible to manually develop search patterns for all 155 methods in PSI-MI, we first observed the distribution of methods in the training data. The 5 most frequently used methods alone form 67.3% of the unique pairs of methods and articles (see table 1). So we decided to focus on these methods for handcrafted patterns, informed by biological insights, and derive the rest of the patterns automatically from PSI-MI by the following process: (A) extraction of names and synonyms from PSI-MI, (B) derivation of patterns by automatic generation of variants by inclusion/deletion of spaces, tabs, newlines, returns, hyphens, etc. and allowing free variation of uppercase and lowercase.

As expected, the results of these automatically generated patterns were bad, especially for precision. Therefore, handcrafted patterns for the most frequent methods[2] were developed by our team's computational linguist and biologist in an iterative process of identifying undetected articles (false negatives), manually finding hints for methods, constructing patterns, and testing them. This process was most successful for MI:0007 (anti tag coimmunoprecipitation), MI:0006 (anti bait coimmunoprecipitation), and MI:0019 (coimmunoprecipitation). As the automatically generated patterns for MI:0096 (pull down) and MI:0018 (two hybrid) were already quite good, the handcrafted patterns did not perform much better. The approach leads to good recall but low precision (R=73.4%, P=24.3%, F=36.5%), over all file-method-pairs in the training data.

As an example of a handcrafted pattern, consider the method MI:0428, which is named "imaging techniques" in PSI-MI 2.5. This name is not actually used by authors, however strings beginning with "colocaliz" or "colocalis" (allowing hyphens and spaces within the string) are a very good indicator for this method.[4]

At this point, rather than focusing on improving the patterns, it was decided to consider the results obtained (methods for a given file) as a set of candidates, which could be filtered with statistical means.[5] A reduction from about 6.8 candidate methods (per file) to about 2.2 (as in the training data) seemed most promising. For this reduction, an empirically derived formula connecting the frequency of the method in the data and the quality of our patterns for this method was

[2]The five methods in table 1 plus MI:0428 (imaging techniques), because of low recall of the automatically generated patterns, and MI:0401 (biochemical), because of low precision.

[3]The results were evaluated by the organizers according to different criteria. We have chosen here the evaluation which corresponds to the approach used to compute the results presented in this paper (aiming at maximizing the F-score)

[4]This pattern could actually be derived from the names of several obsolete precursors (MI:0021, MI:0022, MI:0023) for MI:0428.

[5]Actually the main reason for this is the conceptual difference between "finding every mention of a method" (which our patterns already did with good precision) and finding all interaction detection methods in a file i. e. identifying the methods used by the authors to detect protein-protein interactions. The statistics are a simple way to give more importance to methods which are unlikely to be just mentioned without a connection to the detected interactions.

INTACT	BCMS	OWN	Journal
615	5958	5513	The Journal of biological chem.
280	583	0	Cell
170	1142	910	PNAS
147	1290	931	Molecular and cellular biology
143	1048	804	The EMBO journal
143	572	0	Nature
88	437	0	Science
87	626	0	Biochem. and biophys. res.com.
86	298	0	Molecular cell
75	359	0	Genes & development
58	432	102	Biochemistry
56	527	375	Oncogene
55	261	0	Journal of molecular biology
54	526	445	The Journal of cell biology
...	...	0	...
3260	22804	9080	Total

Table 3: Journal frequencies in INTACT, BCMS and our own dataset

used. For each method M we compute the following weight:

$w_M = f_M * \frac{p_M^2}{r_M^2}$

where f_M is the relative frequency of method M, while p_M and r_M are precision and recall of all patterns for method M.

The candidate methods were ranked according to their weights. We submitted 3 official runs (where the results of IMS were coupled with the results of IPS) and 3 non-official runs (where the results of IMS were not coupled with the results of IPS). Of these runs, **run 1** was maximizing precision (by giving only the best candidate and so hurting recall for all papers containing more than one method), **run 2** was maximizing recall (giving the three best candidates, so hurting precision for all papers containing one or two methods) and **run 3** was maximizing F-score (additional condition that candidates 2 and 3 reached a minimum in frequency and precision). Our best results for the training data and the official runs for the test data of BioCreAtIvE are shown in table 2.

One of the possible criticism to our approach is that the usage of methods might be time-dependent. In other words, it is reasonable to assume that some methods might be frequently used in some periods and then might go 'out of fashion', perhaps because newer and better methods take their place.

3 Evaluation

After the end of BioCreAtIvE the organizers decided to set up a publicly accessible service to give access to some of the systems which performed best in the competition. This work resulted in a meta-server (Leitner et al., 2008), which receives a request from a remote user (either via web interface or via XML-RPC) and forwards the request (via XML-RPC) to specific servers maintained by the participants. The services currently offered by the meta-server are Gene Mention, Gene Normalization, Interaction Article and Taxon Classification. The organizers defined a list of 22804 PubMed papers to be analyzed by each server (which we will call the BCMS dataset).

Interactions	Methods	%
38220	MI:0018 (two hybrid)	25.5
29268	MI:0676 (tap)	19.8
21205	MI:0096 (pull down)	14.4
20509	MI:0397 (two hybrid array)	13.5
12998	MI:0398 (two hybrid pooling)	8.8
11332	MI:0006 (anti bait coip)	7.7
9473	MI:0007 (anti tag coip)	6.4
6331	MI:0399 (2h fragment pooling)	4.3
6089	MI:0363 (inferred by author)	4.1
1842	MI:0004 (affinity chrom)	1.2
...	...	...
147584	total	100%

Table 4: Distribution of methods per interaction in INTACT

Papers	Methods	%
1121	MI:0018 (two hybrid)	34.4
1066	MI:0096 (pull down)	32.7
840	MI:0007 (anti tag coip)	25.8
761	MI:0006 (anti bait coip)	23.4
574	MI:0114 (x-ray diffraction)	17.6
287	MI:0019 (coip)	8.8
251	MI:0416 (fluorescence imaging)	7.7
123	MI:0663 (confocal microscopy)	3.8
120	MI:0424 (protein kinase assay)	3.7
115	MI:0071 (molecular sieving)	3.5
111	MI:0004 (affinity chrom)	3.4
82	MI:0676 (tap)	2.5
...	...	...
3259	total	-

Table 5: Distribution of methods per paper in INTACT[6]

Our initial aim was to offer our IMS tools as an additional service to be integrated in the meta-server, so we started from the BCMS list of articles.We also wanted to be able to test our results against already annotated articles at INTACT (which we will call the INTACT dataset).

The first problem to deal with is that of the format of the input data. Our approach requires the availability of a full-document plain text version of the original article. Initially, we considered using only articles available in PubMed Central, given the standardized XML format which

[6]Notice that one paper can contain multiple methods, so the sum of all values in this table is larger than 100%.

Year	INTACT	INTACT/BCMS	%
1978	1	0	0%
1980	1	0	0%
1987	1	0	0%
1988	4	0	0%
1989	1	0	0%
1990	2	0	0%
1991	2	0	0%
1992	2	0	0%
1993	14	0	0%
1994	21	0	0%
1995	38	4	10.5%
1996	61	11	18.0%
1997	96	25	26.0%
1998	144	33	22.9%
1999	182	54	29.7%
2000	242	58	24.0%
2001	268	67	25.0%
2002	320	80	25.0%
2003	360	64	17.7%
2004	461	66	14.3%
2005	304	50	16.4%
2006	418	131	31.3%
2007	255	6	2.4%
2008	55	0	0%

Table 6: Distribution of INTACT-curated papers per year, and their proportion in the INTACT/BCMS dataset

would definitely simplify conversion to plain text. Unfortunately BCMS has a low overlap with PubMed Central (only 35 articles).

Therefore we decided to implement our own dedicated HTML to text converters for the most frequent journals in BCMS.[7] We focused on journals which appear to have a reasonably standard HTML structure for the articles, and which were easily obtainable from our library service, obtaining a total of 9080 converted articles. Table 3 shows the most frequent journals in INTACT and BCMS, and for each of them the number of articles that we converted ('OWN'). Among the converted articles, 649 are also present in the INTACT set (as of May 31st, 2008). This is the dataset upon which we base our experiments (which in the rest of this paper will be referred to as the INTACT/BCMS dataset).

In INTACT every protein interaction is associated with the papers where it is discussed and with the experimental method that was used to detect it. Table 4 reports the most frequently used methods based on the number of interactions that they are associated with. However, there are some methods which, although used very rarely, can deliver a large number of interactions. One example is MI:0676 (tap), which is used in only 82 papers. In one of them alone (pubmed:16429126) it is associated with 21574 interactions!

[7] Although a generic HTML to text converter could have been used for the application that we describe in this paper, our aim is not only to extract the experimental methods, but also the protein interactions, using a full NLP approach, for which we need a much better conversion.

Year	P (%)	R (%)	F (%)	Big5 (%)
1995	50	66.7	57.2	71
1996	46.9	71.4	56.6	70
1997	47.9	55.7	51.5	68
1998	41.1	55.7	47.3	68
1999	44.9	58.3	50.7	65
2000	47.6	59.6	52.9	69
2001	42.5	58.6	49.3	65
2002	44.2	53.1	48.2	64
2003	44.9	51.6	48.0	62
2004	39.4	48.1	43.3	59
2005	35.7	45.9	40.2	63
2006	33.2	41.6	36.9	60
2007	44.4	61.5	51.6	60
Total	41.2	51.7	45.9	64

Table 7: Performance over INTACT/BCMS data distributed per year of publication. The last column shows the frequency of the 'Big5' experimental methods per year.

Table 5 shows the methods most frequently used, counting only once a method occurring multiple times in the same paper. As our approach delivers the methods per paper (rather than per interaction), these numbers are a more useful guideline to the relative importance of each method.

Using metadata from the corresponding PubMed entry, we get the year of publication of each INTACT paper. Using that information, we can verify how much the methods depend on the year of publication of the paper. Table 6 shows the distribution of INTACT papers by year of publication, and their proportion in our INTACT/BCMS dataset. Despite the relatively recent start of INTACT (2003), the coverage is reasonably good for the years 1997-2007.

Table 7 shows the results of applying the IMS system, as described in the previous section, to the INTACT/BCMS dataset. All tests have been performed using the modality 'max F-score' of the IMS tools, and the results apply to the association article/method (we do not consider yet the association of methods with specific protein interactions). The data provides a sufficiently large time-window, with good distribution for most of the years of observations (with the exceptions of 1995, 2007 and possibly 1996). The results are comparable to those obtained in BioCreAtIvE (both training and test), which are shown in Table 2.

Surprisingly, the value of precision is always

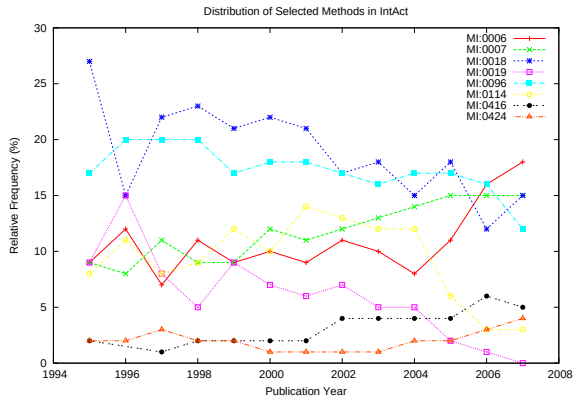


Figure 1: Trends for methods in INTACT

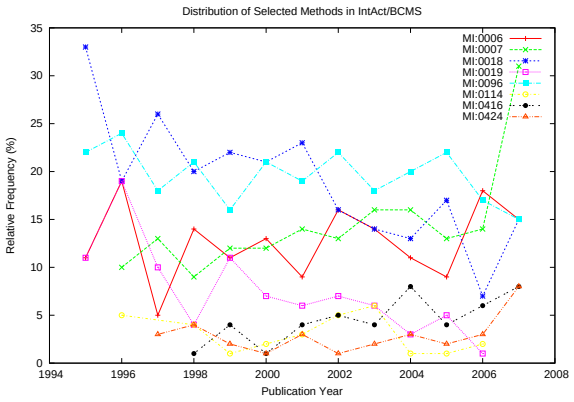


Figure 2: Trends for methods in INTACT/BCMS

lower than recall, which was not expected (in order to maximize the F-score, P and R should be almost equal). This might be caused by the fact that the articles with only one method are more frequent in INTACT (43.2%) than in BioCreAtIvE (34%). There appears to be a decreasing trend in the years 2004 to 2006 (2007 is too small to be representative), which could be caused by the emergence of new experimental methods and reduced usage of methods that were popular in previous years. However, whether this effect is due to a genuine 'aging' of experimental methods, or it is simply due to the selection of articles by INTACT curators, cannot be said on the basis of the available data.

The last column of table 7 shows that the frequency of the Big5 methods declines only slowly, and figure 1 demonstrates that emerging methods, such as MI:0114 (x-ray diffraction), MI:0416 (fluorescence imaging) and MI:0424 (protein kinase assay), take more importance even more hesitantly. Table 7 on the whole confirms that the approach as such seems not endangered by sudden 'revolutions' in the use of experimental methods and a gradual erosion of our results can be contrasted by a periodic reassessment of methods for which handcrafted patterns have to be developed.

4 Discussion

Given the limited set of documents used in our experiments, it is important to ask the question whether the results are sufficiently representative. Since our approach is based upon patterns, each of which is designed to recognize lexical hints to a given experimental method, it is obvious that the approach can be successful only as long as there is no large variation in the relative frequency of methods used in a given set of papers.

4.1 Trends

We have therefore observed the distribution and historical trends of methods in the whole INTACT dataset and compared it with the distribution and historical trends in our own dataset (see figures 1 and 2). Among the 10 most frequently used methods, the number of them shared in both sets is between 6 and 8 for each year. The most frequent methods are the same as in the BioCreAtIvE training data (see table 1). The proportion of these five methods in the INTACT data, distributed per year, is shown in the last column of table 7.

Tables 10 and 11 illustrate the performance of our search patterns for specific methods over the INTACT/BCMS dataset. Of these, the first 5 methods are searched for by handcrafted patterns, the following methods by automatically derived patterns. The disappearance of MI:0019 (coimmunoprecipitation) over time can at least partially be explained by the increase in the use of MI:0007 (anti tag coip) and MI:0006 (anti bait coip). As these are hyponyms of MI:0019, the process we observe may not be an evolution of different scientific practices but actually a semantic process: an increasing preference for the use of a more specific term, be it by the authors of the papers themselves or by curators of INTACT. The identification of 'challengers', i.e. new methods increasing in use (per paper), and so probably deserving handcrafted search patterns, is rather difficult. The most obvious candidate is method MI:0114 (x-ray crystallography), for which the automatically derived pattern does already perform very well, but MI:0416 (fluorescence microscopy) and MI:0424 (protein kinase assay) for

Methods	Papers	%
1	1408	43.2%
2	928	28.5%
3	567	17.4%
4	255	7.8%
5	81	2.5%
6	17	0.5%
7	4	0.1%

Table 8: Number of distinct methods per paper in INTACT

	real			pseudo			oracle
	1	2	3	1	2	3	-
TP	427	705	773	3260	5112	6036	715
FP	222	584	1124	0	1408	3744	642
FN	1069	791	723	3260	1408	484	781
P	65.8	54.7	40.7	100	78.4	61.7	52.7
R	28.5	47.1	51.7	50	78.4	92.6	47.8
F	39.8	56.0	45.6	66.7	78.4	74.1	50.1

Table 9: Comparison of real and simulated experiments

which the performance of the automatically derived patterns is very weak are very promising candidates (see table 11). On a lower level, methods such as MI:0004 (affinity chromatography technology), MI:0047 (far westernblotting) and MI:0071 (molecular sieving) seem to be the most interesting candidates, and the very weak performance for MI:0004 could certainly profit very much from a handcrafted search pattern.

4.2 Independence INTACT/ BioCreative

Since the original program was developed using the BioCreAtIvE training data, it is important to verify that the data on which we are testing do not have a major overlap with BioCreAtIvE training data. Among the whole 'INTACT/BCMS' articles, 453 are not in the BioCreAtIvE training data, 196 are (30.2%). Additionally, 521 of the files BioCreAtIvE training data are not in 'INTACT/BCMS'. The overlap with the BioCreAtIvE test set is less relevant, since it was not used for development of the program, however we report it here to show the independence of the two tests. 522 INTACT/BCMS files are not in BioCreAtIvE-Test, 127 are (19.6%). 231 BioCreAtIvE test files are not INTACT files.

4.3 Choosing the number of methods

The selection of the number of methods for each paper does have an impact on the final results. If the program is set to deliver always only the best ranked method, precision will be relatively good, but recall will be poor. Conversely, if always the 3 best methods are delivered, the opposite will happen. Table 9 shows the results obtained by the system if only the 1st best method is delivered, the 2 best methods, or the 3 best methods ('real').

Another way to observe the impact of the selection of the number of methods on the results is to conduct the following pseudo experiment: suppose we have the perfect ranking algorithm which delivers for each paper a list of all methods correctly ranked for relevance. If, for all papers, we always output only the best method, we are never damaging precision, but we are reducing recall of all but the 1-method-files. If, instead, we take always the two best methods, precision will be lowered by taking many unnecessary 'second best' methods, but we increase recall. Finally, if we decide to assign to all papers the three best methods, precision will be much lower and recall will keep improving. Using the data gathered directly from INTACT we can compute the results, which are also presented in table 9 ('pseudo').

Finally, we can consider the following experiment. Suppose we have an "oracle" which tells us reliably how many methods we should deliver for each paper, how good would be our results? This is a rather realistic scenario, since ideally the method detection program would be coupled with an interaction detection program,[6] therefore knowing how many methods are needed. Although we do not have at the moment a program capable of predicting how many methods should be associated to each paper, we can simulate it with data taken out of INTACT: this will be our "oracle". With such an help, we can filter the results of the method selection and ranking program, obtaining the results that are show in the last column of table 9.

These results show that, although usually our approach delivers the correct ranking for methods, there must be some cases where a correct method is ranked lower than a wrong method. A detailed inspection of these results will provide useful hints for further development.

4.4 Future Directions

The work described in this paper proves that it is possible, with reasonably simple techniques, to capture the most relevant methods with high reliability. Additional improvements to the system

[6] We are developing such a program separately, based on our BioCreAtIvE submission for the PPI-IPS task.

are likely to require complex fine tuning.

As it is impossible to handcraft rules for all the 155 methods, it would be meaningful to investigate how to improve the existing approach via machine learning. To this aim, we performed an experiment with a standard text classifier, using the methods as categories. Although the results were rather disappointing, this might have been due to the poor preparation of the input data. We intend to further investigate if better preprocessing or the usage of more sophisticated classifiers might help overcome these limitations.

The usage of other terminologies/ontologies for the extraction of synonyms (e.g. Mesh) is hampered by the unclear mapping of the relevant entries into PSI-MI entries. Without such a mapping, any attempt at using other dictionary sources would simply increase the level of noise.

Any further evaluation of the results would need to take into account the limitations of the gold standard. If the program finds a method, which has been used by the authors, and it is prominently mentioned in the paper, but it is not included in the gold standard (maybe because it is not directly related to any of the interactions annotated by INTACT curators), then it gets penalized (one FP). As an example, in PubMed 16293613 there are several mentions of "x-ray crystal structure(s)" in connection with the author's experiments, one of these mentions is in the experimental procedures section, which seems to show that method MI:0114 (x-ray crystallography) was used, but this was rated as an FP by comparison with the INTACT gold standard.

As a service to the community, we plan to make available the functionality of method identification as a web service, possibly integrated into the BioCreAtIvE meta-server described in (Leitner et al., 2008). We aim at offering coverage of all PubMed articles for which the full text is freely available, focusing in particular on PubMed Central.

5 Conclusion

We described a system capable of automatically extracting experimental methods for detection of protein interactions from biomedical scientific literature. Participation to the BioCreAtIvE II evaluation has proven the competitiveness of the approach. In this paper we have proven that the range of applicability of the system goes well beyond the scope of the BioCreAtIvE dataset. Reasonable results have been shown over literature spanning the last ten years.

Acknowledgments

We thank the anonymous reviewer for their insightful comments and helpful suggestions. This research is partially funded by the Swiss National Science Foundation (grant 100014-118396/1). Additional support is provided by Novartis Pharma AG, Basel.

References

H Hermjakob, L Montecchi-Palazzi, G Bader, J Wojcik, L Salwinski, A Ceol, S Moore, S Orchard, U Sarkans, C von Mering, B Roechert, S Poux, E Jung, H Mersch, P Kersey, M Lappe, Y Li, R Zeng, D Rana, M Nikolski, H Husi, C Brun, K Shanker, C Grant SG, Sander, P Bork, W Zhu, A Pandey, A Brazma, B Jacq, M Vidal, D Sherman, P Legrain, G Cesareni, I Xenarios, D Eisenberg, B Steipe, C Hogue, and Apweiler R. 2004. The hupo psi's molecular interaction format - a community standard for the representation of protein interaction data. *Nat. Biotechnol*, 22:177–183.

S. Kerrien, Y. Alam-Faruque, B. Aranda, I. Bancarz, A. Bridge, C. Derow, E. Dimmer, M. Feuermann, A. Friedrichsen, R. Huntley, C. Kohler, J. Khadake, C. Leroy, A. Liban, C. Lieftink, L. Montecchi-Palazzi, S. Orchard, J. Risse, K. Robbe, B. Roechert, D. Thorneycroft, Y. Zhang, R. Apweiler, and H. Hermjakob. 2006. IntAct - Open Source Resource for Molecular Interaction Data. *Nucleic Acids Research*.

Martin Krallinger, Florian Leitner, Carlos Rodriguez-Penagos, and Alfonso Valencia. 2008. Overview of the protein-protein interaction annotation extraction task of BioCreative II. *Genome Biology*. (to appear).

Florian Leitner, Martin Krallinger, Carlos Rodriguez-Penagos, Jörg Hakenberg, Conrad Plake, Cheng-Ju Kuo, Chun-Nan Hsu, Richard Tzong-Han Tsai, Hsi-Chuan Hung, William W. Lau, Calvin A. Johnson, Rune Saetre, Kazuhiro Yoshida, Yan Hua Chen, Sun Kim, Soo-Yong Shin, Byoung-Tak Zhang, William A. Baumgartner, Lawrence Hunter, Barry Haddow, Michael Matthews, Xinglong Wang, Patrick Ruch, Frédéric Ehrler, Arzucan Özgür, Günes Erkan, Dragomir R. Radev, Michael Krauthammer, ThaiBinh Luong, Robert Hoffmann, Chris Sander, and Alfonso Valencia. 2008. Introducing meta-services for biomedical information extraction. *Genome Biology*.

Fabio Rinaldi, Thomas Kappeler, Kaarel Kaljurand, Gerold Schneider, Manfred Klenner, Simon Clematide, Michael Hess, Jean-Marc von Allmen, Pierre Parisot, Martin Romacker, and Therese Vachon. 2008. OntoGene in BioCreative II. *Genome Biology*, 9, Suppl 2:S13.

A. Zanzoni, L. Montecchi-Palazzi, M. Quondam, G. Ausiello, M. Helmer-Citterich, and G. Cesareni. 2002. MINT: a Molecular INTeraction database. *FEBS Letters*, 513(1):135–140.

Year	tp	fn	fp	P	R	F
MI:0006 (anti bait coimmunoprecipitation)						
1995	1	0	0	100%	100%	100%
1996	2	2	1	66.7%	50%	57.2%
1997	1	2	5	16.7%	33.3%	22.2%
1998	5	5	5	50%	50%	50%
1999	4	9	12	75%	30.8%	43.7%
2000	7	11	12	36.8%	38.9%	37.8%
2001	4	9	9	30.8%	30.8%	30.8%
2002	10	20	8	55.6%	33.3%	41.7%
2003	5	18	7	41.7%	21.7%	28.5%
2004	7	11	13	35%	38.9%	36.8%
2005	2	8	11	15.4%	20%	17.4%
2006	23	33	27	46.0%	41.0%	43.4%
2007	0	2	0	0	0	0
total	71	130	110	39.2%	35.3%	37.1%
MI:0007 (anti tag coimmunoprecipitation)						
1995	0	0	1	0	0	0
1996	1	1	2	33.3%	50%	40%
1997	2	6	3	40%	25%	30.8%
1998	3	3	8	27.3%	50%	35.3%
1999	10	5	10	50%	66.7%	57.2%
2000	10	7	5	66.7%	58.8%	62.5%
2001	10	10	7	58.8%	50%	54.0%
2002	12	14	11	52.2%	46.2%	49.0%
2003	13	12	9	59.1%	52.0%	55.3%
2004	13	12	9	59.1%	52.0%	55.3%
2005	8	6	6	57.1%	57.1%	57.1%
2006	27	14	19	58.7%	65.9%	62.1%
2007	3	1	1	75%	75%	75%
total	112	91	91	55.2%	55.2%	55.2%
MI:0018 (two hybrid)						
1995	3	0	0	100%	100%	100%
1996	3	1	2	60%	75%	66.7%
1997	15	1	4	78.9%	93.8%	85.7%
1998	13	1	5	72.2%	92.9%	81.3%
1999	27	0	8	77.1%	100%	87.1%
2000	28	0	5	84.8%	100%	91.8%
2001	30	2	9	76.9%	93.8%	84.5%
2002	31	0	13	70.5%	100%	82.7%
2003	23	0	10	69.7%	100%	82.1%
2004	20	0	12	62.5%	100%	76.9%
2005	18	1	7	72%	94.7%	81.8%
2006	20	0	12	62.5%	100%	76.9%
2007	2	0	3	40%	100%	57.1%
total	233	6	90	72.1%	97.5%	82.9%
MI:0019 (coimmunoprecipitation)						
1995	2	0	1	66.7%	100%	80.0%
1996	4	0	5	44.4%	100%	61.5%
1997	4	2	12	25%	66.7%	36.4%
1998	2	1	14	12.5%	66.7%	21.1%
1999	8	5	21	27.6%	61.5%	38.1%
2000	6	3	26	18.8%	66.7%	29.3%
2001	6	3	27	18.2%	66.7%	28.6%
2002	6	7	36	14.3%	46.2%	21.8%
2003	7	2	23	23.3%	77.8%	35.9%
2004	2	3	27	6.9%	40%	11.8%
2005	2	3	21	8.7%	40%	14.3%
2006	0	2	69	0%	0%	0%
2007	0	0	4	0%	0%	0%
total	49	31	286	14.6%	61.3%	23.6%
MI:0096 (pull down)						
1995	1	1	1	50%	50%	50%
1996	4	1	1	80%	80%	80%
1997	9	2	3	75%	81.8%	78.3%
1998	12	3	3	80%	80%	80%
1999	16	3	7	69.6%	84.2%	76.2%
2000	24	4	5	82.8%	85.7%	83.7%
2001	23	3	12	65.7%	88.5%	75.4%
2002	33	9	6	84.6%	78.6%	81.5%
2003	27	2	5	84.4%	93.1%	88.5%
2004	28	4	6	82.4%	87.5%	84.9%
2005	18	6	2	90%	75%	81.8%
2006	43	9	18	70.5%	82.7%	76.1%
2007	2	0	1	66.7%	100%	80.0%
total	240	47	70	77.4%	83.6%	80.4%

Table 10: Most frequent methods (Big5): distribution per year

Year	tp	fn	fp	P	R	F
MI:0114 (x-ray crystallography)						
1995	0	0	2	0	0	0
1996	1	0	0	100%	100%	100%
1997	0	0	1	0	0	0
1998	3	0	1	75%	100%	85.7%
1999	1	0	1	50%	100%	66.7%
2000	2	1	4	33.3%	66.7%	44.4%
2001	3	1	4	42.9%	75%	54.6%
2002	8	2	4	66.7%	80%	72.7%
2003	6	4	1	85.7%	60%	70.6%
2004	0	1	2	0	0	0
2005	0	1	4	0	0	0
2006	2	3	12	14.3%	40%	21.1%
2007	0	0	0	0	0	0
total	26	13	36	41.9%	66.7%	51.5%
MI:0424 (protein kinase assay)						
1995	0	0	0	0	0	0
1996	0	0	0	0	0	0
1997	0	2	0	0	0	0
1998	0	3	0	0	0	0
1999	0	2	1	0	0	0
2000	0	1	1	0	0	0
2001	2	2	0	100%	50%	66.7%
2002	0	2	2	0	0	0
2003	0	4	2	0	0	0
2004	1	4	0	100%	20%	33.3%
2005	0	2	0	0	0	0
2006	0	10	2	0	0	0
2007	1	0	0	100%	100%	100%
total	4	32	8	33.3%	11.1%	16.6%
MI:0004 (affinity chromatography technology)						
1995	0	1	0	0	0	0
1996	0	1	0	0	0	0
1997	0	0	0	0	0	0
1998	0	1	1	0	0	0
1999	0	2	0	0	0	0
2000	0	2	0	0	0	0
2001	0	4	1	0	0	0
2002	0	4	1	0	0	0
2003	0	1	0	0	0	0
2004	1	1	2	33.3%	50%	40%
2005	0	4	1	0	0	0
2006	0	4	1	0	0	0
2007	0	0	0	0	0	0
total	1	25	7	12.5%	3.8%	5.8%
MI:0047 (far western blotting)						
1995	0	1	0	0	0	0
1996	0	0	0	0	0	0
1997	2	2	0	100%	50%	66.7%
1998	0	4	0	0	0	0
1999	1	2	0	100%	33.3%	50%
2000	0	1	0	0	0	0
2001	1	2	1	50%	33.3%	40%
2002	0	0	0	0	0	0
2003	0	4	0	0	0	0
2004	0	3	0	0	0	0
2005	0	2	0	0	0	0
2006	1	2	0	100%	33.3%	50%
2007	0	0	0	0	0	0
total	5	23	1	83.3%	17.9%	29.5%
MI:0071 (molecular sieving)						
1995	0	0	0	0	0	0
1996	0	0	0	0	0	0
1997	0	0	0	0	0	0
1998	0	0	2	0	0	0
1999	0	1	1	0	0	0
2000	0	2	2	0	0	0
2001	1	1	2	33.3%	50%	40%
2002	0	2	2	0	0	0
2003	2	4	3	40%	33.3%	36.3%
2004	2	2	3	40%	50%	44.4%
2005	0	2	4	0	0	0
2006	2	2	7	22.2%	50%	30.7%
2007	0	0	0	0	0	0
total	7	16	26	21.2%	30.4%	25.0%

Table 11: Other important methods: distribution per year

Semantic MEDLINE: A Web Application for Managing the Results of PubMed Searches

Halil Kilicoglu[1,2], Marcelo Fiszman[1], Alejandro Rodriguez[1], Dongwook Shin[1], Anna M. Ripple[1] and Thomas C. Rindflesch[1]

[1] National Library of Medicine, Bethesda, MD, 20892, USA
[2] Concordia University, Department of Computer Science and Software Engineering, Montreal, QC, H3G 1M8, Canada

Abstract

We describe Semantic MEDLINE, a Web application that manages the results of PubMed searches by summarizing and visualizing semantic predications extracted from MEDLINE citations and linking them to several structured resources to provide an integrated environment. To demonstrate its utility, we present a scenario in which we use Semantic MEDLINE to gain insights into relaxin, a hormone whose function in humans has not been fully elucidated. We propose Semantic MEDLINE as an enabling information resource and exploration tool for biomedical scientists, health care professionals, and consumers. (For access, send e-mail to trindflesch@mail.nih.gov).

1 Introduction

Traditional information retrieval tools often challenge users with the large number of items returned. In the biomedical domain, PubMed provides access to over 17 million citations from some 5000 journals in the MEDLINE database. Sophisticated knowledge management applications are needed to help the user exploit this massive amount of text. Similarly, the amount of structured online health-related information, including biomedical vocabularies, ontologies, clinical and molecular biology knowledge bases, and model organism annotation databases, is growing at a rate that outpaces the development of effective access applications.

Linking the biomedical literature and structured resources presents new opportunities in user-driven text mining and knowledge discovery as well as automatic curation of biomedical resources. We are developing a Web application, called Semantic MEDLINE, which integrates PubMed with natural language processing, automatic summarization, visualization, and interconnections among multiple sources of relevant biomedical information. The system is intended to help health care professionals and consumers keep abreast of current research as well as assist researchers in mining the literature to generate hypotheses. In this paper, we first describe Semantic MEDLINE and its implementation and then discuss a scenario using the tool to elucidate the peptide hormone relaxin.

2 Related Work

Natural language processing often underpins applications in biomedicine, and some systems extract relations from text (Blaschke *et al.*, 1999; Friedman *et al*., 2001; Leroy *et al*., 2003; Lussier *et al*., 2006; Rindflesch and Fiszman, 2003). Others focus on using the information extracted; examples include automatic summarization (McKeown *et al.*, 2001, Fiszman *et al*., 2004a), question answering (Demner-Fushman and Lin, 2007; Jacquemart and Zweigenbaum, 2003; Sable *et al*., 2005; Sneiderman *et al.*, 2007; Wedgwood, 2005), and literature-based knowledge discovery (Ahlers *et al*., 2007b; Hristovski *et al.,* 2006; Srinivasan and Libbus, 2004; Swanson, 1986).

Several recent systems visualize the information extracted. Ali Baba (Plake *et al*., 2006) relies on concepts co-occurring in documents to represent text as a graph of interrelated relationships. Based on co-occurrences of genes in MEDLINE abstracts, Jensen *et al.* (2001) construct networks of genes found relevant in gene expression data analysis. The Telemakus project (Fuller *et al*., 2004) is based on relationships identified by hand and is meant to enable knowledge discovery through interactive visual maps of linked concepts among documents. The LitMiner system (Feldman *et al.*, 2003) represents several gene-related relations extracted with a type of underspecified natural language processing in a graph. Finally, the PGViewer tool (Tao *et al.*, 2005) visualizes genomic information across both structured and textual databases. Integrating

the biomedical literature with external databases and ontologies has also been explored: GoPubMed (Doms and Schroeder, 2005) and CiteXplore (http://www.ebi.ac.uk/citexplore).

3 Background

At the core of Semantic MEDLINE are two existing tools: SemRep (Rindflesch and Fiszman, 2003), which extracts semantic predications (subject-predicate-object triples) from text, and an automatic summarizer (Fiszman *et al.,* 2004a).

3.1 SemRep

SemRep was developed for the biomedical research literature and uses domain knowledge provided by the Unified Medical Language System (UMLS) (Lindberg *et al.*, 1993). It represents textual content with semantic predications consisting of UMLS Metathesaurus concepts as arguments and UMLS Semantic Network relations as predicates. Processing relies on an underspecified syntactic analysis based on the SPECIALIST Lexicon (McCray *et al.*, 1994) and MedPost part-of-speech tagger (Smith *et al.*, 2004). MetaMap (Aronson, 2001) maps simple noun phrases to Metathesaurus concepts, and "indicator rules" map syntactic elements to Semantic Network predicates. For example, SemRep identifies the three semantic predications in (2) from the sentence fragment in (1):

(1) … dexamethasone is a potent inducer of multidrug resistance-associated protein expression in rat hepatocytes …

(2) Dexamethasone STIMULATES Multidrug Resistance Associated Proteins
Multidrug Resistance-Associated Proteins PART_OF Rats
Hepatocytes PART_OF Rats

These predications comprise executable knowledge and are amenable to further automatic manipulation.

3.2 Automatic Summarization

In the semantic abstraction paradigm of automatic summarization (Hahn and Mani, 2000) semantic predications serve as representation of the source text and are manipulated to generate a salient overview of input text. SemRep predications from multiple documents provide input to the Semantic MEDLINE summarizer, which provides a reduced and focused list of predications (a "semantic condensate").

Semantic condensates are based on a user-selected topic and a summarization perspective (Treatment of Disease, Substance Interactions, Diagnosis, or Pharmacogenomics). Each perspective is represented as a set of formal constraints on the arguments and the predicate of the input predications.

In all perspectives, the transformation from the initial list of predications to the reduced list in the semantic condensate is guided by four principles, which are informally defined as:

- *Relevance*: Include predications on the topic of the summary that conform to the selected summarization perspective
- *Connectivity*: Include additional useful predications on the basis of having shared arguments with the "relevant" predications
- *Novelty*: Eliminate, using UMLS hierarchical information, the predications the user already knows, identified as those having generic arguments, such as "Pharmaceutical Preparations" or "Disease"
- *Saliency*: Eliminate predications with low frequency of occurrence

4 System Implementation

4.1 Enhancing SemRep

SemRep had originally been developed with an emphasis on clinical research; it was enhanced for Semantic MEDLINE to accommodate linking the research literature to structured resources, including genetic databases. SemRep now augments mappings provided by MetaMap with ABGene (Tanabe *et al.*, 2002) and pattern matching to recognize and normalize gene names to Entrez Gene (Maglott *et al*., 2007). For example, MetaMap is unable to map the token "c-Jun" to a Metathesaurus concept; however, ABGene identifies it as a gene, and the normalization routine maps it to the Entrez Gene official symbol "JUN" and records its gene identifier (3725). The normalization mechanism uses a pre-computed index based on Entrez Gene official symbols, names, and aliases stored in a Berkeley DB table. The normalization index is updated periodically and is currently limited to human genes.

4.2 Semantic MEDLINE

Semantic MEDLINE is implemented as a three-tier, Java EE-based Web application (Fig. 1), which allows separation of user interface, application logic, and data storage. We leverage ma-

ture open-source technologies to the extent possible. The application runs in a Tomcat servlet container on an Apache http server and has been developed using the Apache Struts Web application framework (http://struts.apache.org/). This encourages the use of the MVC (Model-View-Controller) paradigm to provide a clean separation of application model, navigational code, and page design code through the use of Java Servlet API.

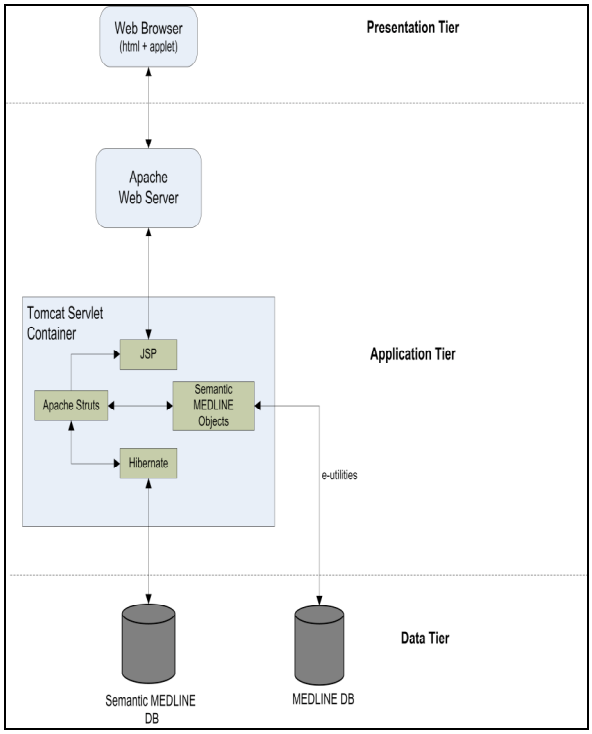


Fig. 1. Semantic MEDLINE architecture

A MySQL database is used to store Semantic MEDLINE data, which includes semantic predications extracted from MEDLINE citations in addition to UMLS Metathesaurus and Entrez Gene data. The database tables are pre-populated from plain text files that contain SemRep output and Metathesaurus/Entrez Gene data using Perl scripts. The Hibernate object/relational mapping (ORM) tool (http://www.hibernate.org/) provides enhanced database access through database connection pooling and query caching.

Semantic MEDLINE supports PubMed searching through NCBI's Entrez Programming Utilities API (http://eutils.ncbi.nlm.nih.gov/) to provide real-time access to PubMed records, retrieved and manipulated in XML format.

To visualize the semantic condensates as graphs in Semantic MEDLINE, we developed a Flash application using the Adobe Flex framework (http://www.adobe.com/products/flex) and the Flare visualization toolkit (http://flare.prefuse.org/), the ActionScript extension of the Prefuse toolkit written in Java. Nodes in a graph represent arguments in SemRep predications, and the arcs predicates. We enhanced the visualization capabilities provided by Flare by linking the semantic predications in the graph to external structured biomedical resources.

Arcs are linked to the MEDLINE citations from which the corresponding predications were extracted, while nodes are linked to three resources in addition to Entrez Gene: the UMLS Semantic Navigator (Bodenreider, 2000), Online Mendelian Inheritance in Man (OMIM) (Hamosh *et al.*, 2002), and Genetics Home Reference (GHR) (Mitchell *et al.*, 2004).

Linking to the UMLS Semantic Navigator uses Metathesaurus concept identifiers (CUI) and allows the user to view the context of a predication argument in the UMLS hierarchy. The gene name normalization procedure discussed above enables linking to Entrez Gene. OMIM identifiers are extracted from the OMIM *morbidmap* file periodically and associated with UMLS Metathesaurus concepts in the Semantic MEDLINE database, while GHR identifiers are extracted from GHR XML files periodically and, similarly, associated with Metathesaurus concepts.

SemRep is not fast enough to accommodate Semantic MEDLINE in real time. We therefore run SemRep on the MEDLINE database in an off-line process and store the extracted predications in the MySQL database as they become available. Currently, the database contains 9,224,765 predications extracted from 2,779,669 citations processed by MEDLINE during 2004, 2005, 2006, and 2007.

5 User Interface

The Semantic MEDLINE Web page has four tabs: Search, SemRep, Summarization and Visualization. The Search tab allows the user to specify a query and select PubMed limits. Titles of retrieved citations are displayed in tabular format, hyperlinked to PubMed. On this page and throughout, Semantic MEDLINE results can be saved in XML format for later reuse.

The SemRep tab presents predications extracted from citations retrieved. The user can then move to the Summarization tab and select a topic and perspective. Topics appear in a drop-down list, sorted by frequency of occurrence in the underlying SemRep predications.

Fig 2. A view from the Summarization tab

The user may also choose to disable filtering based on frequency of occurrence of predications (saliency filter). Fig. 2 shows a view from the Summarization tab.

The Visualization tab provides access to the graph representing the summarized semantic condensate, which guides navigation through the documents retrieved by the search. Nodes and arcs are color coded according to meaning. Node colors are determined by UMLS semantic groups (e.g. substances, procedures, disorders) (McCray *et al.*, 2001). The color legends for the nodes and arcs are displayed in the Filters tab on the right pane. Each item in the legends is a check box, and clicking on one of them shows or hides the nodes (or arcs) with that semantic type (or predicate) in the graph, providing focused views.

Clicking on a graph element displays information in the Information tab on the right pane. In addition to frequency of occurrence of the corresponding argument or predication, information for nodes includes UMLS concept identifier and semantic type for the corresponding argument as well as links (if available) to external resources, including the UMLS Semantic Navigator, Entrez Gene, GHR, and OMIM; for arcs, arguments of the corresponding predication and predicate name are given. The Citation button enables viewing the MEDLINE citations from which the predication was extracted, including PubMed identifier, title and abstract. The citation sentence in which the predication is asserted is highlighted. (See Fig.3 for some aspects of the visualization)

6 Evaluation

We have so far not conducted a user-centered evaluation. Accuracy of the predications generated by SemRep is crucial to overall effectiveness of Semantic MEDLINE. A summary of prior evaluations of SemRep and the automatic summarizer (see Table 1) suggests that average precision is near 77%. The evaluations conducted have generally been post-hoc and considered precision only; one study also assessed recall.

In each study, evaluation was limited to particular predicates: hypernymic (ISA) relations (Rindflesch and Fiszman, 2003), gene-disease etiological relations, such as CAUSE and PREDISPOSE, (Rindflesch *et al.,* 2003b) and finally, those relations focusing on pharmacogenomics, such as DISRUPTS and INHIBITS (Ahlers *et al.,* 2007a).

Evaluation of the automatic summarizer involved assessing accuracy of the predications in semantic condensates produced from various summarization perspectives. Two focused on treatment of disease (Fiszman *et al.*, 2004a; Fiszman *et al.*, 2004b), one with MEDLINE citations, and the other with an online medical encyclopedia as source documents. Semantic condensates regarding drug information were also evaluated (Fiszman *et al.*, 2006). All evaluation results are presented in Table 1.

7 Investigating Relaxin

We describe a scenario exploiting the components of Semantic MEDLINE to elucidate relaxin, a peptide hormone originally connected

Evaluation type and reference	Number of predications	Precision	Recall
SemRep			
Hypernymic (Rindflesch and Fiszman, 2003)	830	83%	
Gene-disease (Rindflesch *et al.*, 2003b)	1124	76%	
Pharmacogenomics (Ahlers *et al.*, 2007a)	623	73%	55%
Automatic summarizer			
Treatment of disease (Fiszman *et al.*, 2004a)	306	66%	
Treatment of disease (Fiszman *et al.*, 2004b)	190	87%	
Drug information (Fiszman *et al.*, 2006)	189	78%	
Total	**3262**	**77%**	

Table 1. SemRep/automatic summarization evaluation results

with parturition and more recently found to have a wider range of physiological implications. On the Search page, the user issues the query "relaxin" to PubMed, with the default dates 01/01/2004 through 12/31/2007 reflecting the part of MEDLINE currently available for processing. From PubMed Limits, accessible under "More options," "Abstracts" is selected. This query retrieves 349 citations, which generate 2899 predications (on the SemRep page). On the Summarization page the user chooses "Substance Interactions" as Summary Type and "Relaxin" as Summary Topic. The Saliency Filter (keeping only the most frequent predications) yields 119 predication tokens.

Summarized predications are displayed on the Visualization page as a graph, which provides an informative overview of the characteristics of relaxin as extracted from the retrieved citations. The user can also follow links to retrieve more detailed information on selected aspects of the graph. Contributing resources are the citations (linked to graph arcs) that produced the predications as well as related citations computed by PubMed. Additional structured knowledge sources include the UMLS Metathesaurus GHR, OMIM, and Entrez Gene (linked to graph nodes).

The current graph consists of 21 predication types with four predicate types: ISA, CAUSES, AFFECTS, and INTERACTS_WITH. One predication is disconnected from the main graph ("Isoproterenol CAUSES myocardium; injury"); the other 20 are connected with "Relaxin" as the central concept.

Hierarchical structure in the Metathesaurus, accessible from graph nodes, provides general information about the entities that relaxin is involved with. For example, two of these are shown to be peptides:

- Angiotensin II → Angiotensins → peptide hormone
- Adenylate Cyclase → Intracellular Signalling Peptides and Proteins → Peptides

Perusal of predicate types in the graph elucidates the major characteristics of relaxin in a principled way. "Relaxin" is in the following relationships:

- ISA: Hormones, peptide hormone
- CAUSES: Premature Birth
- AFFECTS: Renal fibrosis, Contraction, Apoptosis, Hemodynamics
- INTERACTS_WITH: Angiotensin II, Collagen, Progesterone, Adenylate Cyclase, Interleukin-11, RXFP1, RXFP2

Concentrating first on the ISA predications (extracted from 52 citations) provides an overview of relaxin function. For example, the first citation accessible from the arc between "Relaxin" and "Hormones" indicates an important relaxin function "…reverses cardiac and renal fibrosis…" (PMID 15967869), while the fourth (PMID 17266534) is a review article describing other relaxin characteristics: "…denoted initially as a hormone of pregnancy…" and "…many other physiological roles have been identified for relaxin, including cardiovascular and neuropeptide functions and an ability to induce the matrix metalloproteinases…" Further exploration of the graph reveals additional aspects of relaxin's activities. For example, clicking on the arc (ISA) between "Relaxin" and "peptide hormones" reveals a cognitive function for relaxin. The title of the first citation (PMID 16262650) is "Relaxin

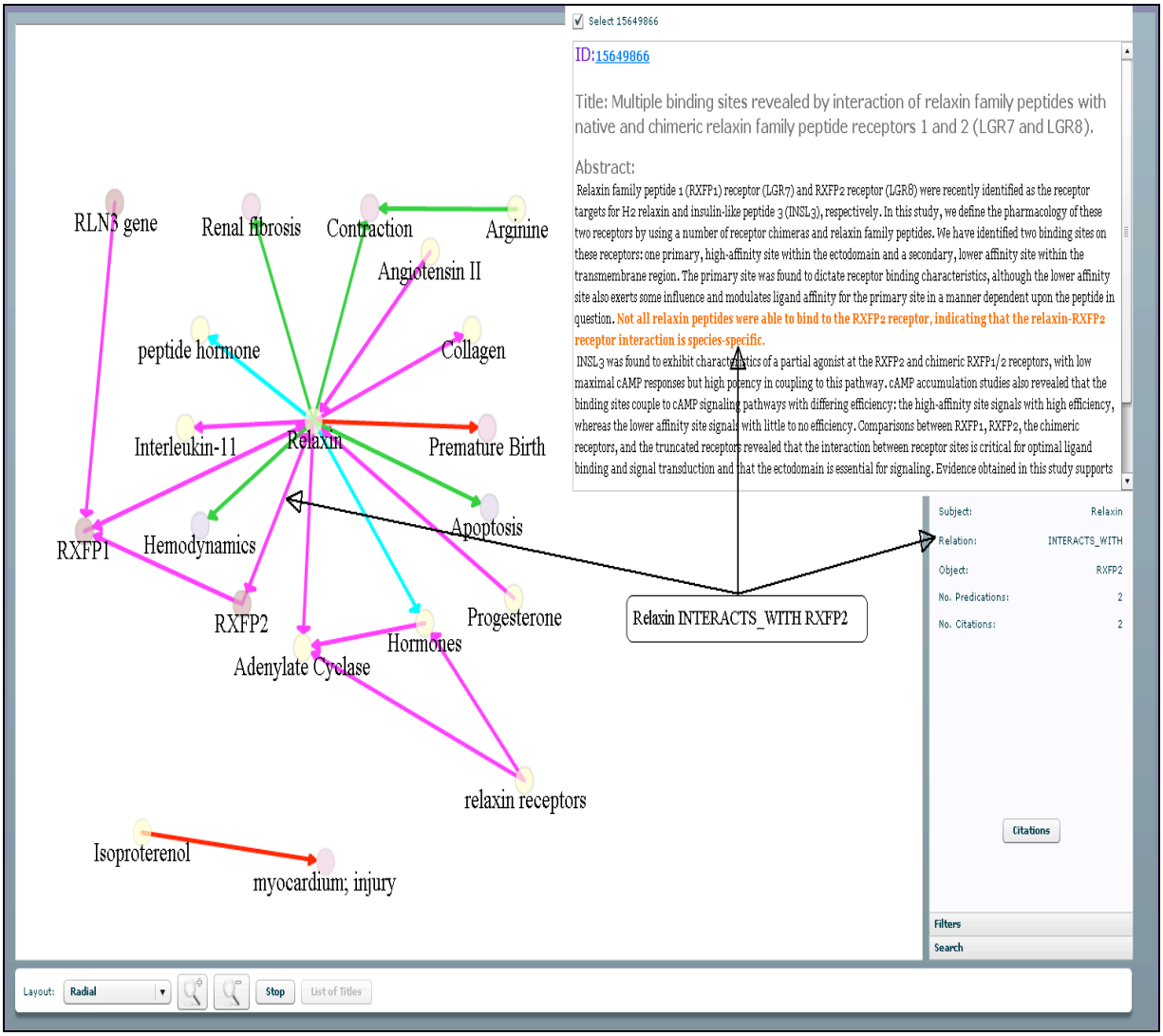


Fig. 3. Visualizing summarization results for Relaxin search, with Relaxin INTERACTS_WITH RXFP2 relation highlighted.

receptor activation in the basolateral amygdala impairs memory consolidation. ”

Information associated with the graph allows the user to pursue some particular aspect of relaxin in greater detail. For example, there are five citations available by clicking on the AFFECTS arc between “Relaxin” and “Hemodynamics.” Based on the known effects of relaxin during pregnancy, some of the basic research reported in these citations investigates its properties more generally. One of them (PMID 15198972), for example, tested “…whether relaxin can modify systemic arterial hemodynamics and load when chronically administered to nonpregnant rats,” while the goal of another (PMID 16172427) was “to determine the cardiovascular effects of rhRLX in hypertensive rats.” Another study (PMID 15271674) suggests practical implications: “…we speculate about the therapeutic potential of relaxin in renal and cardiovascular diseases.”

As noted above, SemRep precision is around 80%. A SemRep error in the graph is “relaxin receptors INTERACTS_WITH Hormones,” which was incorrectly extracted from two citations (PMID 14965317 and 15240635). Although neither asserts this predication, both publications may nonetheless be of interest regarding relaxin function. The title of the first is “Relaxin: new functions for an old peptide” and that of the second is “Increased expression of the relaxin receptor (LGR7) in human endometrium during the secretory phase of the menstrual cycle.”

The graph also serves as a guide to investigating the underlying mechanisms of relaxin. Interaction with two genes, RXFP1 and RXFP2, is shown. The title of one of the citations (PMID 15649866) that generated the predication assert-

ing interaction with RXFP2 confirms that these are the two major receptors for relaxin: "Multiple binding sites revealed by interaction of relaxin family peptides with native and chimeric relaxin family peptide receptors 1 and 2 (LGR7 and LGR8)."

Further exploration of RXFP2 is possible in Entrez Gene, which is accessible through a direct link from the RXFP2 node. Entrez Gene provides a wealth of technical information about this gene and its associated protein, including aliases (LGR8; GREAT; GPR106; INSL3R; LGR8.1; RXFPR2) and a brief summary. The functional information in the summary is augmented by Gene Ontology annotations and GeneRIFs (gene references into function), which are curated descriptive phrases culled from relevant MEDLINE citations. Finally, Entrez Gene provides links to a large number of structured knowledge sources, such as HGNC (HUGO Gene Nomenclature Committee) and KEGG (Kyoto Encyclopedia of Genes and Genomes).

Fig. 3 shows the visualization for the Relaxin search. One of the citations from which the predication "Relaxin INTERACTS_WITH RXFP2" is generated (PMID 15649866) is displayed, with the sentence that generates the predication highlighted.

8 Conclusion

We discussed the Semantic MEDLINE Web application, which helps PubMed users manage search results based on semantic natural language processing, automatic summarization, and visualization. To show its utility, we used the application as a guide in examining the peptide hormone relaxin, whose functions and mechanisms are not fully understood.

We are currently in the process of semantically analyzing the MEDLINE database and scaling the system without compromising performance. As the knowledge sources we rely on, including the UMLS and Entrez Gene, are continually updated, one challenge is to keep relevant data up-to-date. In addition, a large number of citations are added to MEDLINE daily, and these need to be made available through Semantic MEDLINE. At this time, we are putting in place procedures to automate data updating.

We are also exploring the extension of Semantic MEDLINE to supporting additional health-related textual databases, such as *ClinicalTrials.gov*. Finally, we plan to formally evaluate the user interface, which will no doubt lead to reassessing some of our design decisions and ultimate improvements in overall effectiveness of the application.

Acknowledgment

This research was supported in part by the Intramural Research Program of the National Institutes of Health, National Library of Medicine.

References

Ahlers, C., Fiszman, M., Demner-Fushman, D., Lang, F.-M., and Rindflesch. T. (2007a). Extracting semantic predications from Medline citations for pharmacogenomics. In *Proceedings of Pacific Symposium on Biocomputing,* 12:209-20.

Ahlers, C., Hristovski, D., Kilicoglu, H., and Rindflesch, T.C. (2007b) Using the Literature-Based Discovery Paradigm to Investigate Drug Mechanisms, In *Proceedings of AMIA Annual Symposium*, pp.6-10.

Aronson, A.R. (2001) Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. In *Proceedings of AMIA Annual Symposium,* pp. 17-21.

Blaschke, C., Andrade, M., Ouzounis, C., and Valencia, A. (1999) Automatic extraction of biological information from scientific text: protein-protein interactions, In *Proceedings of the 7th International Conference on Intelligent Systems for Molecular Biology*. pp. 60-7.

Bodenreider, O. (2000) A semantic navigation tool for the UMLS. In *Proceedings of AMIA Fall Symposium,* pp. 971.

Demner-Fushman, D., Lin, J. (2007) Answering clinical questions with knowledge-based and statistical techniques. *Computational Linguistics,* 33 (1):63-103.

Doms, A. and Schroeder, M. (2005) GoPubMed: exploring PubMed with the Gene Ontology. *Nucleic Acids Research*, 33(Web Server issue):W783-6.

Feldman, R., Regev, Y., Hurvitz, E., and Finkelstein-Landau, M. (2003) Mining the biomedical literature using semantic analysis and natural language processing techniques. *Biosilico*, 1(2):69-80.

Fiszman, M., Rindflesch, T.C., and Kilicoglu, H. (2004a) Abstraction summarization for managing the biomedical research literature. In *Proceedings of HLT-NAACL Workshop on Computational Lexical Semantics.* pp. 76-83.

Fiszman, M., Rindflesch, T.C., and Kilicoglu, H. (2004b) Summarization of an online medical encyclopedia. *MEDINFO*, 506-10.

Fiszman, M., Rindflesch, T.C., and Kilicoglu, H. (2006) Summarizing drug information in MEDLINE citations. In *Proceedings of AMIA Annual Symposium,* pp. 254-8.

Friedman, C., Kra, P., Yu, H., Krauthammer, M., and Rzhetsky, A. (2001) GENIES: a natural language processing system for the extraction of molecular pathways from journal articles. *Bioinformatics*, 17(Supplement 1):S74-82.

Fuller, S.S., Revere, D., Bugni, P., and Martin, G.M. (2004) A knowledgebase system to enhance scientific discovery: Telemakus. *Biomedical Digital Libraries*, 1(1):2.

Hamosh, A., Scott, A.F., Amberger, J., Bocchini, C., Valle, D., and McKusick, V.A. (2002) Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders. *Nucleic Acids Research*, 30(1):52-5.

Hahn, U. and Mani, I. (2000) The challenges of automatic summarization. *Compute*r, 33(11):29-36.

Hristovski, D., Friedman, C., Rindflesch, T.C., and Peterlin, B. (2006) Exploiting semantic relations for literature-based discovery. In *Proceedings of AMIA Annual Symposium,* pp. 347-53.

Jacquemart, P. and Zweigenbaum, P. (2003) Towards a medical question-answering system: a feasibility study. *Studies in Health Technology and Informatics*, 95:463-8.

Jensen, T.K., Laegreid, A., Komorowski, J., and Hovig, E. (2001) A literature network of human genes for high-throughput analysis of gene expression. *Nature Genetics*, 28(1):21-8.

Leroy, G., Chen, H. and Martinez, J.D. (2003) A shallow parser based on closed-class words to capture relations in biomedical text. *Journal of Biomedical Informatics*, 36(3):145-58.

Lindberg, D. A., Humphreys, B. L., and McCray, A. T. (1993) The Unified Medical Language System. *Methods of Information in Medicine.* 32(4):281–91.

Lussier, Y., Borlawsky, T., Rappaport, D., Liu, Y., and Friedman C. (2006) PhenoGO: assigning phenotypic context to Gene Ontology annotations with natural language processing. In *Proceedings of Pacific Symposium on Biocomputing*, pp. 64-75.

Maglott, D., Ostell, J., Pruitt, K.D., and Tatusova, T. (2007) Entrez Gene: gene-centered information at NCBI. *Nucleic Acids Research*, 35(Suppl):D26-31.

McCray, A.T., Burgun, A., and Bodenreider, O. (2001). Aggregating UMLS semantic types for reducing conceptual complexity. *MEDINFO*, 10(Pt 1):216-20.

McCray, A.T., Srinivasan, S., and Browne, A.C. (1994) Lexical methods for managing variation in biomedical terminologies. In *Proceedings of Annual Symposium on Computer Applications in Medical Care*, pp. 235-9.

McKeown, K.R., Chang, S.-F., Cimino, J., Feiner, S., Friedman, C., Gravano, L., Hatzivassiloglou, V., Johnson, S., Jordan, D.A., Klavans, J. L., Kushniruk, A., Patel, V., and Teufel, S. (2001) PERSIVAL, a system for personalized search and summarization over multimedia healthcare information. In *Proceedings of the ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 331-40.

Mitchell, J. A., Fun, J., and McCray, A.T. (2004) Design of Genetics Home Reference: A new NLM consumer health resource. *Journal of the American Medical Informatics Association.* 11(6):439-47.

Plake, C., Schiemann, T., Pankalla, M., Hakenberg, J., and Leser, U. (2006) ALIBABA: PubMed as a graph. *Bioinformatics*, 22(19): 2444-5.

Rindflesch, T. C., and Fiszman, M. (2003) The interaction of domain knowledge and linguistic structure in natural language processing: Interpreting hypernymic propositions in biomedical text. *Journal of Biomedical Informatics,* 36(6):462-77.

Rindflesch, T.C., Libbus, B., Hristovski, D., Aronson, A.R., and Kilicoglu, H. (2003) Semantic relations asserting the etiology of genetic diseases. In *Proceedings of AMIA Annual Symposium,* pp. 554-8.

Sable, C., Lee, M., Zhu, H.R., and Yu, H. (2005) Question analysis for biomedical question answering. In *Proceedings of AMIA Annual Symposium*, pp. 1102.

Smith, L., Rindflesch, T.C., and Wilbur, W.J. (2004) MedPost: a part-of-speech tagger for biomedical text. *Bioinformatics,* 20(14):2320-1.

Sneiderman, C.A., Demner-Fushman, D., Fiszman, M., Ide, N., and Rindflesch, T.C. (2007) Knowledge-based methods for helping clinicians find answers in MEDLINE. *Journal of the American Medical Informatics Association*, 14(6):772-80.

Srinivasan, P. and Libbus, B. (2004) Mining MEDLINE for implicit links between dietary substances and diseases. *Bioinformatics*, 20(Suppl 1):I290–I296.

Swanson, D.R. (1986) Fish oil, Raynaud's syndrome, and undiscovered public knowledge. *Perspectives in Biology and Medicine*, 30(1):7-18.

Tanabe, L., Wilbur, W.J. (2002) Tagging gene and protein names in biomedical text. *Bioinformatics*, 18(8):1124-32.

Tao, Y., Friedman, C., and Lussier, Y.A. (2005) Visualizing information across multidimensional post-genomic structured and textual databases. *Bioinformatics*, 21(8):1659-67.

Wedgwood, J. (2005) MQAF: a medical question-answering framework. In *Proceedings of AMIA Annual Symposium*, pp. 1150.

Extracting Protein-Protein Interactions from Text using Rich Feature Vectors and Feature Selection

Sofie Van Landeghem, Yvan Saeys and Yves Van de Peer
Department of Plant Systems Biology, VIB, 9052 Gent, Belgium
Department of Molecular Genetics, Ghent University, 9052 Gent, Belgium
yves.vandepeer@psb.ugent.be

Bernard De Baets
Department of Applied Mathematics, Biometrics and Process Control,
Ghent University, 9000 Gent, Belgium

Abstract

Because of the intrinsic complexity of natural language, automatically extracting accurate information from text remains a challenge. We have applied rich feature vectors derived from dependency graphs to predict protein-protein interactions using machine learning techniques. We present the first extensive analysis of applying feature selection in this domain, and show that it can produce more cost-effective models. For the first time, our technique was also evaluated on several large-scale cross-dataset experiments, which offers a more realistic view on model performance.
During benchmarking, we encountered several fundamental problems hindering comparability with other methods. We present a set of practical guidelines to set up a meaningful evaluation.
Finally, we have analysed the feature sets from our experiments before and after feature selection, and evaluated the contribution of both lexical and syntactic information to our method. The gained insight will be useful to develop better performing methods in this domain.

1 Introduction

Results of genetic studies are published on a daily basis and appear in scientific articles, accessible through online literature services like PubMed (http://pubmed.gov). Over 17 million citations are currently available through PubMed and this resource is still growing exponentially. Fully automated systems that extract biological knowledge from text have thus become a necessity.

Many approaches have been proposed to extract biological information from research articles. The first methods mainly relied on co-occurrence of biological entities. They would classify 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). Typically, a co-occurrence based technique exhibits high recall, but low precision.

A second important set of techniques apply patterns or rules which are usually hand-crafted, allowing the method to obtain high precision while recall typically drops. The RelEx system uses three rules in combination with information derived from dependency graphs (Fundel et al., 2007). Dependency parsing uses graph topology to represent syntactic relations between individual words of the sentence (Figure 1).

Finally, machine learning techniques use training data to construct a model, which is then applied to a test set to predict protein-protein interactions (PPIs). To extract meaningful features for the model construction, dependency parsing is often used. Both global context, such as the root of the tree, and local context, such as the parent of a particular node, can be taken into account (Katrenko and Adriaans, 2007). Erkan et al. (2007) extract sentences where two proteins co-occur with an interaction word. Extracted features include the interaction words themselves and the parents of the proteins in the dependency graph. Kim et al. (2008) present a walk kernel, consisting of patterns of two vertices and their intermediate edge (*vertex-walk* or *v-walk*), as well as sequences of two edges and their common vertex (*edge-walk* or *e-walk*), extracted from the shortest path between two proteins in the graph. They also conclude that a feature-based approach performans better than direct kernel techniques. A

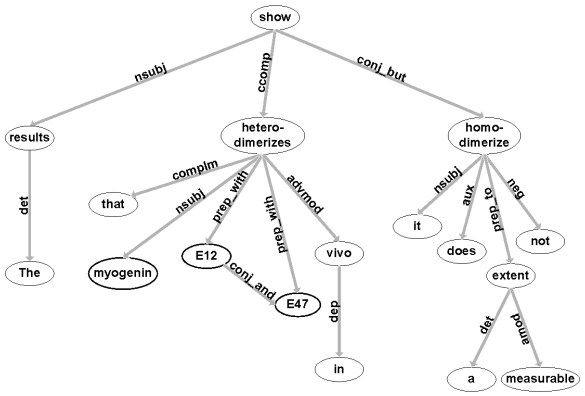


Figure 1: The dependency graph for the sentence 'The results show that myogenin heterodimerizes with E12 and E47 in vivo, but it does not homodimerize to a measurable extent'

more reduced feature set is used by Fayruzov et al. (2008), taking mainly syntactic information into account. Finally, deep syntactic parsing can be combined with a shallow dependency parser to create a more accurate model (Saetre et al., 2008).

A hybrid approach is also possible, with handcrafted rules forming the basis for different kernels, which are then aggregated by linear combination (Giuliano et al., 2006).

The application of feature selection in the domain of natural language processing is relatively new. Jiang and Zhai (2007) have investigated the type of features that are potentially useful for relation extraction in general. Feature selection techniques have also been employed for the task of text classification (Wang et al., 2008). However, to the best of our knowledge, this paper presents the first study of applying rich feature vectors in combination with feature selection for protein-protein interaction extraction. 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). Furthermore, it is the first time that a broad cross-corpus study has been conducted, offering an evaluation in a more realistic setup than cross-validation on a single dataset.

2 Benchmarking protein-protein interaction extraction techniques

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 (Van Landeghem et al., 2008). In this section we will analyse these problems and introduce practical guidelines in order to improve comparability between extraction methods in this domain.

2.1 Benchmark datasets

The development of standard benchmarking datasets is a step forward towards meaningful comparisons between different information extraction techniques. For genetic interaction extraction, such corpora include AIMed (Bunescu et al., 2005), Bioinfer (Pyysalo et al., 2007), HPRD50 (Fundel et al., 2007), IEPA (Ding et al., 2002) and LLL (Nedellec, 2005). These corpora all have slightly different scopes, ranging from protein-gene interactions concerned with *Basilicus subtilis* transcription to human protein-protein interactions. Recently, conversion software has been introduced to convert these different datasets into a common data format, forming a rich corpus with a broad range of genetic interactions (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., 2008). It has been shown by Pyysalo et al. (2008) that the choice of benchmark dataset can drastically influence extraction performance. It is therefor advisable to evaluate algorithms on a collection of different corpora.

2.2 Instance extraction

Even when evaluating on the same dataset, different preprocessing steps can yield a varying set of instances. Homodimers, which are self-interacting proteins, are sometimes discarded from the dataset. A similar problem is raised by nested annotations in the corpus which may or may not be discarded, influencing the final number of instances in the dataset. To construct negative examples, the closed world assumption is usually adapted, stating that no interaction exists between two entities when there is no annotated evidence. We believe it is best to always clearly indicate which rules were applied for instance extraction, and to report on the number of retrieved instances.

2.3 The extraction task

For the extraction of protein-protein interactions, it is often assumed that the proteins in the text are known a priori. However, when performing the named entity recognition (NER) step automatically, errors will propagate and cause a drop in performance. We believe that the NER step is a different subtask which should be examined and evaluated separately. Similarly, parse trees can be automatically constructed or manually verified. In our opinion, parsing input sentences in a fully-automated fashion is necessary to provide a scalable method, applicable to large datasets.

2.4 Cross validation

Ideally, abstracts for the testing phase should be completely hidden during training. Saetre et al. (2008) pointed out that 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. 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. It is therefore best to modify the regular cross-validation approach to include all instances from one sentence in the same fold, or even define folds consisting of complete abstracts.

2.5 Counting true positives

Finally, the definition of a true positive is ambiguous in the text-mining domain. Each pair of proteins is usually considered as an individual instance, evaluated independently of others. However, two distinct instances may be expressing the same interaction. Thus, to extract a true protein-protein interaction, retrieving one such instance suffices. The latter evaluation technique naturally exhibits higher recall. Even though this technique is useful for benchmarking complete information retrieval systems, we feel that instance-level evaluation is more representative for the task of extracting interactions between named entities from individual sentences.

3 Methods

In our study, we used all the datasets that have been converted to a common data format by Pyysalo et al. (2008), with the exception of Bioinfer. This corpus is relatively new, and contains extensive annotations of proteins and interactions. For example, the words *alpha 5 integrins* are annotated as being a protein reference in the construct *alpha 5 and beta 1 integrins*. Our extraction method assumes a protein is mentioned as a contiguous stream of tokens, which are replaced by the token *PROT* for all training and testing instances in the dataset. This is why we exclude Bioinfer from further analysis and focus on the other four corpora: AIMed, HPRD5, IEPA and LLL. However, we plan on resolving these issues in the future, as well as considering more corpora to test our method on, such as theBiocreative and Genia datasets.

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 1: Number of instances in the four corpora

3.1 Dataset preprocessing

In preparing the datasets we excluded homodimers, as not all corpora support homodimer annotation. 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 1.

3.2 Extracting rich feature vectors

Our feature extraction method uses syntactic and lexical patterns derived from the shortest path between two proteins in the dependency graph. These graphs are built automatically using the Stanford parser (de Marneffe et al., 2006). The shortest path in the graph is scanned for all subsequent vertices and their intermediate edge (*v-walk*), as well as all subsequent edges and their common vertex (*e-walk*), taking into account both syntactic and lexical properties of the walks (Table 2, upper four rows). To traverse this path,

Type	Features
Lex v-walk	heterodimer nsubj PROT, heterodimer prep PROT
Syn v-walk	VBZ nsubj PROT, VBZ prep PROT
Lex e-walk	nsubj heterodimer prep
Syn e-walk	nsubj VBZ prep
BOW	PROT, 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 2: Syntactic and lexical features for the pair of proteins (Myogenin, E12) from Figure 1

we go up from the first protein to the root by inverting the original direction of the edges, and go down again from the root to the second protein. To improve generalization of lexical information by the classifier, we apply the Porter stemming algorithm (Porter, 1980). Protein names are substituted by the token *PROT* to enable the classifier to learn interaction patterns, disregarding the specific proteins involved. The walk features are augmented with a bag-of-words (BOW) approach in combination with the stemming algorithm, to capture critical information outside the shortest path of the dependency graph (Table 2, fifth row). This bag-of-words approach will give rise to quite some irrelevant features, which is one of the reasons why we will apply fully automated feature selection techniques after feature extraction. Syntactic and lexical information from the root node are stored as separate features (Table 2, last two rows). Finally there is a numeric feature indicating the length of the shortest path.

All features are encoded by defining one specific numeric feature for each syntactic or lexical pattern, storing 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. For example, when using cross-validation on the AIMed dataset, which is the richest corpus of the four, over 14.000 numeric features are extracted from the training set.

3.3 Classification model

For our experiments, we made use of a linear support vector machine classifier (SVM, 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). We used the Weka[1] implementation of LibSVM, with an internal 5-fold cross-validation loop on the training portion of the data to determine the optimal C-parameter.

3.4 Feature selection

Feature selection (FS) techniques 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 techniques only select 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 will 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 filter method we 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

$$IG(S, D) = H(S) - H(S|D)$$

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

$$= H(S) - \sum_{j \in V(D)} \frac{|S_j|}{|S|} H(S_j)$$

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, and 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.

3.5 Evaluation strategy

For benchmarking our PPI extraction method, we use instance-level evaluation. We have applied regular 10-fold cross-validation (*Instance CV*), as well as the modified version of 10-fold cross validation, with folds consisting of complete abstracts (*Abstract CV*). 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. As a performance measure, the F-measure is used, which is common practice in this domain. It is defined as the harmonic mean between precision (p), which expresses how many of the predictions are correct, and recall (r), which expresses how many of the true interactions are correctly predicted.

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 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, this is the first time such a large-scale cross-dataset comparison has been conducted.

	Corpus	p	r	F
Inst. CV	AIMed	0.66	0.58	0.62
	HPRD50	0.71	0.71	0.71
	IEPA	0.74	0.69	0.71
	LLL	**0.79**	**0.84**	**0.82**
Abstr. CV	AIMed	0.49	0.44	0.46
	HPRD50	0.60	0.51	0.55
	IEPA	0.64	0.70	0.67
	LLL	**0.72**	**0.73**	**0.73**
Co-occ.	AIMed	0.18	1.00	0.30
	HPRD50	0.38	1.00	0.55
	IEPA	0.41	1.00	0.58
	LLL	**0.50**	1.00	**0.66**

Table 3: Evaluation on the four individual datasets

4 Results

4.1 Individual dataset evaluation

As a baseline, we evaluated our method on all datasets separately, using 10-fold instance CV (Table 3, first row). We then evaluated the method using the modified version of 10-fold CV, clustering instances originating from the same sentence in the same fold (Table 3, second row). For the evaluation on AIMed, the original abstract splits were used (Bunescu et al., 2005). We noticed an artificial boost of performance of up to 0.16 F-measure when using instance CV. In both experiments we find a significant difference in F-measure between the best results (LLL) and the worst results (AIMed), ranging between 0.20 and 0.27 F-measure. To demonstrate the inherent differences between the four individual datasets, we have included the results for a simple co-occurrence based technique, assigning a

	Method	p	r	F
AIMed abstr. cv	Rich features	0.49	0.44	0.46
	Fundel et al. (2007)	0.40	0.50	0.44
	Giuliano et al. (2006)	0.61	**0.57**	**0.59**
	Saetre et al. (2008)	**0.64**	0.44	0.52
AIMed inst. cv	Rich features	0.66	0.58	0.62
	Erkan et al. (2007)	0.60	0.61	0.60
	Fayruzov et al. (2008)	0.41	0.50	0.45
	Katrenko and Adriaans (2007)	0.45	**0.68**	0.54
	Saetre et al. (2008)	**0.78**	0.63	**0.70**
LLL inst. cv	Rich features	0.79	0.84	**0.82**
	Fayruzov et al. (2008)	0.72	**0.86**	0.78
	Fundel et al. (2007)	**0.85**	0.79	**0.82**

Table 4: Comparison to existing techniques for individual datasets.

	features	p	r	F	syn	lex	bow
AIMed	14.000	**0.49**	**0.44**	**0.46**	15	61	20
	10.000	0.48	0.43	0.45	16	61	19
	7.500	0.41	0.41	0.41	17	61	18
	5.000	0.44	0.38	0.41	16	59	21
HPRD50	2.600	0.60	0.51	0.55	21	44	29
	1.500	0.51	0.60	0.55	23	48	23
	750	0.57	0.61	0.59	23	52	20
	500	**0.61**	**0.62**	**0.61**	23	45	28
	250	0.58	0.36	0.45	23	51	23
IEPA	6.900	**0.64**	0.70	0.67	17	49	30
	5.000	0.61	0.71	0.65	14	43	38
	2.500	0.63	**0.75**	**0.68**	22	51	21
	1.000	0.54	0.66	0.60	20	42	34
LLL	1.600	0.72	0.73	**0.73**	22	44	28
	800	**0.75**	0.71	**0.73**	27	48	19
	400	0.68	**0.77**	**0.73**	33	44	18
	200	0.54	0.66	0.60	35	58	3

Table 5: FS on individual datasets, showing the distribution of the three most important type of features in percentages (syntactic walks, lexical walks and BOW-features). Evaluation using Abstract CV.

test	features	p	r	F	syn	lex	bow
AIMed	11.300	0.27	0.67	0.38	12	57	28
	10.000	0.27	**0.69**	**0.39**	12	55	28
	7.500	**0.28**	0.65	**0.39**	13	60	24
	5.000	0.27	0.60	0.37	14	61	21
HPRD50	26.100	0.62	**0.52**	0.57	9	67	21
	20.000	0.69	0.51	**0.59**	9	66	21
	15.000	0.76	0.47	0.58	9	66	22
	10.000	**0.80**	0.25	0.38	7	69	20
IEPA	22.500	**0.87**	**0.27**	**0.41**	10	67	21
	20.000	0.84	0.24	0.38	9	66	21
	15.000	0.84	0.23	0.36	10	66	22
	10.000	0.71	0.16	0.25	11	63	23
LLL	26.700	0.54	0.32	**0.40**	9	67	21
	25.000	0.51	**0.33**	**0.40**	8	66	22
	20.000	0.43	0.21	0.28	9	66	22
	15.000	0.53	0.28	0.37	9	66	22
	10.000	**0.93**	0.15	0.26	7	69	20

Table 6: FS on cross-dataset experiments, showing the distribution of the three most important type of features in percentages (syntactic walks, lexical walks and BOW-features). Evaluation using three datasets as training data and one dataset as test set.

true interaction between each co-occurring pair of proteins. These results exhibit a difference in F-measure of up to 0.36 between AIMed and LLL.

Subsequently, we compared our method using rich feature vectors to other, recently introduced PPI extraction techniques. To allow for a fair comparison, we only consider studies using a similar evaluation setup. The results of this analysis are shown in Table 4. We observe that our method is comparable to state-of-the art performance, and that it achieves particularly good results when using regular CV on the LLL dataset.

4.1.1 Feature selection

Because our extraction method results in high-dimensional, sparse feature vectors, we have investigated the usability of feature selection techniques to improve performance and obtain faster models. The results of these experiments on the individual datasets are shown in Table 5. On HPRD50, recall could be increased with 0.11 resulting in an increase in F-measure of 0.06, 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 clearly 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 0.05. However, the time necessary to build the classifier for all ten folds is reduced from 6 hours and 5 minutes to 3 hours and 22 minutes, including the FS step itself. This illustrates the usefulness of feature selection to create more cost-effective models.

Analysing the distribution of feature types before and after FS, we see that in general, syntactic features take up a slightly bigger proportion after filtering, usually accompanied by a reduction of word features (Table 5, last three columns). However, lexical information still takes up the biggest part of the feature set.

4.2 Cross dataset experiments

To assess the performance of our method in a more realistic setup, we have conducted large-scale cross-datasets experiments. For this purpose, we used one dataset for testing, and the other three for training, which will cause less bias to a specific training set. These experiments provide an estimate of the out-of-domain generalization ability, by analysing the artificial boost in performance when only performing a single-domain evaluation. It is the first time such a broad cross-corpus study is conducted.

The results of our experiments are shown in Table 6. We see that testing on HPRD50 achieves the best performance, with 0.62 precision, 0.52 recall and 0.57 F-measure. For this corpus, the

test set	Features	p	r	F
AIMed	all	0.27	0.67	**0.38**
	syntactic	**0.28**	0.58	0.37
	lexical	0.24	**0.72**	0.36
HPRD50	all	0.62	**0.52**	**0.57**
	syntactic	**0.70**	0.48	**0.57**
	lexical	0.60	0.50	0.54
IEPA	all	**0.87**	**0.27**	**0.41**
	syntactic	0.62	0.26	0.37
	lexical	0.82	0.17	0.29
LLL	all	0.54	**0.32**	0.40
	syntactic	**0.64**	0.30	**0.41**
	lexical	0.47	0.28	0.35

Table 7: Cross-dataset experiments using lexical information, syntactic information or both

performance is similar to the single-dataset evaluation. 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).

The cross-dataset experiments give rise to high-dimensional datasets, with up to 26.700 features. We have applied FS in order to filter out irrelevant features, and obtain faster models with less risk of overfitting. The results for all four test cases can be found in Table 6. In most cases, we are able to reduce the feature set significantly without loss of performance. Testing on HPRD50, we achieve a gain in precision of 0.14 while only sacrificing 0.05 recall, when the feature set is reduced to 57% of its original size. Model construction with the entire feature set took one hour and 36 minutes, while the classifier was built after 57 minutes using the reduced feature set. The FS step itself only took an additional 5 minutes. This clearly shows that FS can lead to faster and more cost-effective models.

Testing on HPRD50, precision can rise to 0.84 when even more features are filtered, though recall starts dropping at this point. Nevertheless this faster model may be preferred by a user who only wants to extract highly reliable data. On LLL we also obtain much higher precision when sacrificing recall. When using AIMed as test set, we are able to maintain good results when more than half of the features are filtered out.

4.2.1 Contribution of lexical and syntactic information

In order to gain deeper insight into the importance of certain features, we performed a statistical analysis of the contribution of different categories of features (Table 6, last three columns). In general, we saw that 85-90 % of the features consist of lexical information (lexical walks and BOW features combined). This distribution is roughly maintained after feature selection. This indicates that both lexical and syntactic information are important when extracting protein-protein interactions. We validated this assumption by running the cross-dataset experiments again, once with only lexical information, and once with only syntactic information. The results are shown in Table 7, demonstrating that the global performance of both lexical and syntactic approaches are similar to each other. However, when using only syntactic information and comparing this approach to the full feature set, a gain of precision of up to 0.10 can be achieved (HPRD50, LLL), while producing a similar F-score. The only exception to this general rule seems to be when IEPA is used as testing set. In this particular case, high precision is achieved by mainly lexical information. 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 stated by Fayruzov et al. (2008) that using only syntactic information leads to classifiers that are able to perform well, while being independent of a specific lexicon. To improve recall however, including lexical information might still be useful.

5 Conclusions and future work

We have developed a technique to extract protein-protein interactions using rich feature vectors and machine learning techniques. For the extraction of relevant features, semantic information from dependency graphs was used, as well as lexical information from the sentence expressing an interaction. We have discussed some important issues for benchmarking extraction techniques, and have indicated practical guidelines for setting up a meaningful evaluation. 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 se-

lection techniques to show these can improve the generalization performance and 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.

Beyond the approach of rich feature vectors and feature selection, we would like to use the insight gained from these experiments to develop more specific kernel-based approaches for the extraction of protein-protein interactions from text, building further on relation extraction kernels already developed (Kim et al., 2008).

Acknowledgments

SVL would like to thank the Special Research Fund (BOF) for funding her research. YS would like to thank the Research Foundation Flanders (FWO) for funding his research.

References

B. Boser, I. Guyon and V.N. Vapnik. 1992. A training algorithm for optimal margin classifiers. *Proceedings of COLT 1992*, 144-152

R. Bunescu, R. Ge, R. Kate, E. Marcotte, R. Mooney, A. Ramani and Y. Wong. 2005. Comparative Experiments on Learning Information Extractors for Proteins and their Interactions. *Artificial intelligence in medicine*, 33(2):139-155

J.G. Caporaso, N. Deshpande, J.L. Fink, P.E. Bourne, K.B. Cohen and L. Hunter. 2008. Intrinsic evaluation of text mining tools may not predict performance on realistic tasks. *Proceedings of PSB'08*, 640-51

J. Ding, D. Berleant, D. Nettleton and E. Wurtele. 2002. Mining MEDLINE: abstracts, sentences, or phrases? *Proceedings of PSB'02*, 326-337

G. Erkan, A. Ozgur and D. R. Radev. 2007. Extracting interacting protein pairs and evidence sentences by using dependency parsing and machine learning techniques. *Proceedings of BioCreAtIvE 2*

T. Fayruzov, M. De Cock, C. Cornelis and V. Hoste. 2008. DEEPER: a Full Parsing based Approach to Protein Relation Extraction. *Lecture Notes In Computer Science*, 4973, 36-47

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

C. Giuliano, A. Lavelli and L. Romano 2006. Exploiting Shallow Linguistic Information for Relation Extraction from Biomedical Literature. *EACL*

I. Guyon and A. Elisseeff 2003. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3 : 11571182.

L. Hirschman, A. Yeh, C. Blaschke and A. Valencia. 2005. Overview of BioCreAtIvE: critical assessment of information extraction for biology. *BMC Bioinformatics*, 6(Suppl 1):S1

R. Hoffmann and A. Valencia. 2004. A Gene Network for Navigating the Literature. *Nature Genetics*, 36:664

J. Jiang and C. Zhai. 2007. A Systematic Exploration of the Feature Space for Relation Extraction. *Proceedings of NAACL-HLT 07*, 113-120

S. Katrenko and P. Adriaans. 2007. Learning Relations from Biomedical Corpora Using Dependency Trees. *Lecture notes in Computer Science*, KDECB, volume 4366

J.-D. Kim, T. Ohta and J. Tsujii. 2008. Corpus annotation for mining biomedical events from literature. *BMC Bioinformatics*, 19(Suppl 1):i180-i182

S. Kim, J. Yoon and J. Yang. 2008. Kernel approaches for genic interaction extraction. *Bioinformatics*, 9:10

MC. de Marneffe, B. MacCartney and C. D. Manning. 2006. Generating Typed Dependency Parses from Phrase Structure Parses. *Proceedings of LREC'06*

C. Nedellec. 2006. Learning language in logic - genic interaction extraction challenge. *Proceedings of LLL'05*, 31-37

M. F. Porter. 1980. An algorithm for suffix stripping. *Program*, 14(3), 130-137

S. Pyysalo, F. Ginter, J. Heimonen, J. Björne, J. Boberg, J. Järvinen and T. Salakoski. 2007. BioInfer: A corpus for information extraction in the biomedical domain. *BMC Bioinformatics*, 8(50)

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

D. Rebholz-Schuhmann, H. Kirsch, M. Arregui, S. Gaudan, M. Riethoven and P. Stoehr. 2006. EBIMed - text crunching to gather facts for proteins from Medline. *Bioinformatics*, 23(2):e237-e244.

R. Saetre, K. Sagae and J. Tsujii. 2008. Syntactic features for protein-protein interaction extraction. *Proceedings of LBM'07)*

Y Saeys, I. Inza and P. Larrañaga. 2007. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507-2517.

S. Van Landeghem, Y. Saeys, B. De Baets and Y. Van de Peer. 2008. Benchmarking machine learning techniques for the extraction of protein-protein interactions from text. *Proceedings of Benelearn'08*, 79-80.

H. Wang, M. Huang, S. Ding and X. Zhu. 2008. Exploiting and integrating rich features for biological literature classification. *BMC Bioinformatics*, 9(Suppl 3):S4

A Tool for the Automatic and Manual Annotation of Biomedical Documents

Anália Lourenço[1], Sónia Carneiro[1], Rafael Carreira[2], Miguel Rocha[2], Isabel Rocha[1], Eugénio Ferreira[1]
[1] IBB - Institute for Biotechnology and Bioengineering, Center of Biological Engineering
[2] Department of Informatics / CCTC
University of Minho
Campus de Gualtar, 4710-057 Braga – PORTUGAL
{analia,soniacarneiro,
ecferreira,irocha}@deb.uminho.pt,
{rafaelcc,mrocha}@deb.uminho.pt

Abstract

The techniques developed within the field of Biomedical Text Mining (BioTM) have been mainly tested and evaluated over a set of known corpora built by a few researchers with a specific goal or to support scientific competitions. The generalized use of BioTM software therefore requires that an enlarged set of corpora is made available covering a wider range of biomedical research topics. This work proposes a software tool that facilitates the task of building a BioTM corpus by providing a user-friendly and interoperable tool that allows both automatic and manual annotation of biomedical documents (supporting both abstracts and full text). This tool is also integrated in a more comprehensive BioTM framework.

1 Introduction

Semantic annotation, sometimes called concept matching in the biomedical literature, is the process of mapping phrases within a source text to distinct concepts defined by domain experts.

Traditionally, such annotation was exclusively manual. However, the growing scientific publication rate, the continuous evolving of biological terminology and the more complex analysis requirements brought by systems-level approaches urge for automated curation processes (Ananiadou et al., 2006; Natarajan et al., 2005; Erhardt et al., 2006).

The research field of BioTM emerged from this need and has been providing for helpful computerised approaches. In particular, Biomedical Named Entity Recognition (BioNER), the field that deals with the unambiguous identification of named entities (such as names of genes, proteins, gene products, organisms, drugs, chemical compounds, etc.), is the key step for accessing and integrating the information stored in the literature (Zweigenbaum et al., 2007; Jensen et al., 2006; Natarajan et al., 2005).

Techniques for term identification are becoming widely used in biomedical research. Lexical resources (Fundel and Zimmer, 2006; Mukherjea et al., 2004; Kou et al., 2005; Muller et al., 2004) and rule-based systems (Hu et al., 2005; Hanisch et al., 2005) deliver some degree of automation. On the other hand, Machine Learning contributions (Okazaki and Ananiadou, 2006; Kou et al., 2005; Shi and Campagne, 2005; Yeganova et al., 2004; Sun et al., 2006) address issues like term novelty, synonymy (including term variants and abbreviations) and homonymy.

Despite current achievements, technique development and usage are constrained by the limited availability of high-quality training corpora. In fact, at this point, biomedical annotated corpora represent a bottleneck in the development of BioTM software. Existing approaches cannot be extended without the production of corpora, conveniently validated by domain experts.

In this work, a contribution to tackle this matter is provided, with the development of a novel interoperable and user-friendly software application that supports manual curation of biomedical documents. The proposed software implements a workflow where a biomedical corpus is automatically annotated based on a specialised dictionary. The discovered biomedical concept output is then directed into a manual curation stage, and finally a high-quality biomedical annotated corpus is made available.

Both the automatic and manual annotation tasks are envisioned to be flexible, allowing the tagging of many biological entity classes and the creation and use of different dictionaries, extracted from major biomedical databases. Although we have our own annotation schema, the software is expected to be useful within other domains which have domain-specific resources available. In other words, if a new annotation schema is defined and the dictionary builders cope with it, both automatic and manual annotation are granted.

The remainder of this paper starts by placing annotation tools within BioTM scenario, establishing basic requirements and identifying related work. The enumeration of the software development aims follows. Next, the main features of the proposed software application are discussed, namely the creation of particular dictionaries, the default annotation schema, the automatic annotation module and user-friendly manual annotation environment. Final remarks provide an overall perspective of the work and identify new features.

2 The Role of Annotation Tools in BioTM

Emerging efforts in BioTM agree on considering manually annotated biomedical corpora as priceless resources (Kim et al., 2008; Kim et al., 2003). Many researchers openly contribute and disseminate annotated corpora such as GENIA (Kim et al., 2003), PennBioIE (Kulick S et al., 2004) or GENETAG (Tanabe et al., 2005). Also, there are datasets coming from knowledgeable challenges such as BioCreAtive[1]. Yet, adaptation of available resources to new problems (real-world scenarios) usually requires substantial efforts, since they have been designed to meet a particular aim and tend not to comply with any common data format.

The construction of a new corpus implies the laborious and time-consuming manual collection and annotation of a significant number (typically hundreds) of documents. It is not straightforward to gather, organise and annotate a valuable set of documents. On the one hand, the set of documents has to be representative of the domain it is supposed to describe, i.e., it has to embrace the terminological trends that characterise the domain, while establishing a contrast towards other domains. On the other hand, annotation has to be as comprehensible and consensual as possible. According to a given annotation schema, different annotators should be able to agree, producing similar outputs. Otherwise, either the annotation schema is not able to reflect the domain conveniently, or the domain requires further annotation rules that prevent contradicting or misleading outputs.

It is not reasonable to acknowledge the need for corpora without devising computational annotation tools. There exist several manual text annotation tools for creating annotated corpora. General-purpose annotation tools such as Callisto[2], WordFreak[3](Morton and LaCivita, 2003), the General Architecture for Text Engineering (GATE[4]) (Cunningham et al., 2002) and MMAX2[5] are references in the area. However, these tools present limited flexibility and its 'out of the box' usage often demands expert programming skills.

Although offering customisable tasks (for example, a simple annotation schema can be defined with an XML DTD), these tools do not offer any support for biology-related natural language processing. Dedicated tools such as POS taggers, parsers and named entity recognisers are becoming widely available and it would be desirable to include them into annotation tools.

Tools should support semantic annotation by hand and some form of automatic annotation (using available resources such as dictionaries, ontologies, templates or user-specified rules). Moreover, by supporting both syntactic and semantic annotation, a wide variety of annotation schemas can be defined and used. New annotation tasks can be built without writing new software or creating specialised configuration files.

3 Development Aims

[1] http://biocreative.sourceforge.net/

[2] http://callisto.mitre.org/

[3] http://wordfreak.sourceforge.net/

[4] http://gate.ac.uk/

[5] http://mmax.eml-research.de/

The development of our biomedical annotation tools was driven by two important needs, essential for creating useful text corpora: i) accuracy and consistency of the annotations, and ii) usability of the data. The major aim of this work is therefore two-fold: i) to provide a friendly environment for curators and ii) to take advantage of the multiple informational resources available, enhancing the annotation process as much as possible.

In this sense, the baseline requirements of our tools were interoperability with other tools/modules and flexibility in terms of annotation schemas and data exchange formats. Annotation schemas should be made as general as possible, covering major biomedical classes and thus, enabling (partial) schema interchange. Also, document annotation may comprise both syntactic (POS information) and semantic annotations (BioNER information).

The main aim of the annotation environment presented here is to provide common text processing modules and to enable automatic and manual document annotation. The text processing pipeline was modelled with minimal assumptions on their dependences and application ordering. Tokenisation, sentence splitting and stopword removal are the basic text processing steps, and typically do not rely on previous preprocessing, whereas chunk parsing as well as BioNER may be based on POS annotation. Not only the tools should be able to deal with multilayer annotation, as annotation processes should not have precedence over one another, i.e. semantic annotation may occur after or before POS tagging.

Furthermore, neither automatic nor manual annotation processes are considered mandatory. Typically, manual annotation is time-consuming and should be considered a later step, accounting for false positive matches (term homonymy) and miss annotations (term synonymy and term novelty). However, it is up to the user to decide whether to trigger one or the two processes.

4 Implementation

The implementation of our tools devised the following components/modules:

- an input/output module enabling the conversion of documents for common file formats (such as PDF and HTML) to plain text;
- a pre-processing module embracing XML-based text structuring (the title, authors, journal, abstract and the location of major sections are tagged), tokenisation and stopword removal;
- a default annotation schema embracing all major biological entity classes (genes, proteins, compounds and organisms) and some uncommon, although valuable classes (laboratory techniques and physiological states);
- a lexicon-based biomedical annotator which supports the construction of customised dictionaries as well as user-defined rules and lookup tables;
- an user-friendly annotation viewer based on Cascade Style Sheets (CSS) that allows the user to verify and correct annotations and refine dictionary contents.

Additionally, it is important to note that unlike many previous approaches our tools are able to handle both abstracts and full text documents indistinctively. The latter will undoubtedly give an increasing amount of useful information in most cases.

4.1 Lexical Resources

The tool supports two kinds of lexical resources: lookup tables and dictionaries. The authors have prepared lookup lists of standard laboratory techniques and general physiological states. Also, the user may create general or particular dictionaries from major biomedical databases such as BioCyc[6], UniProt[7] or ChEBI[8] and integrated databases such as Biowarehouse[9] (Figure 1). Each data source is characterised in terms of the embraced biological classes and organism (if it is a multi-organism source). The user may decide to include all contents or select just a few, depending on the purpose of the dictionary.

Database copyrights are preserved as there is no content distribution with the tool. In order to deploy any loader, the user has to download the contents from the corresponding source.

[6] http://biocyc.org/
[7] http://www.uniprot.org/
[8] http://www.ebi.ac.uk/chebi/
[9] http://biowarehouse.ai.sri.com/

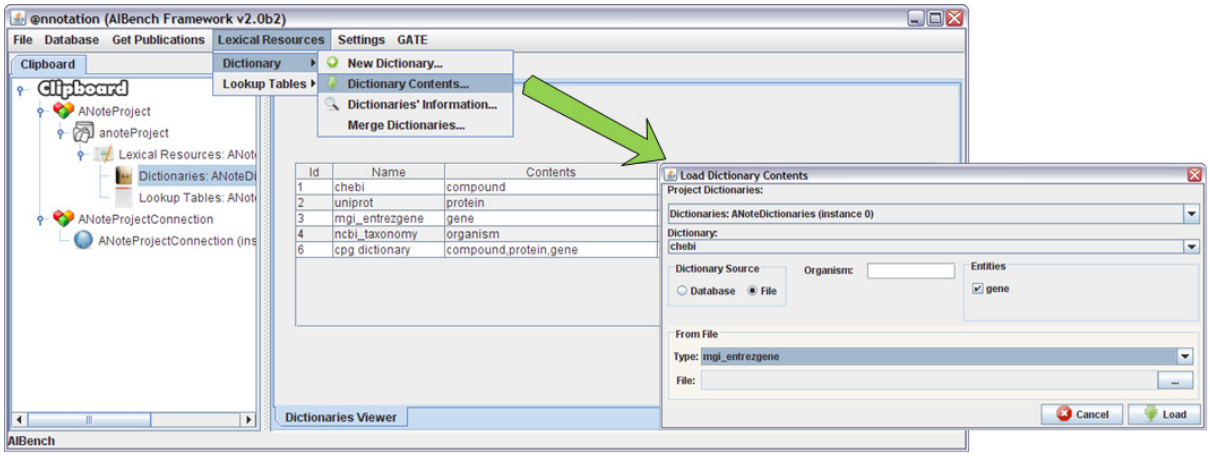


Figure 1. Deploying the construction of a new dictionary using available data loaders.

On the other hand, all created resources are kept in relational format (currently, on MySQL database engine) and thus, allow eventual sharing.

4.2 Annotation Schemas

The default semantic annotation schema was created by the authors and aims at tracking down major biological entities. Currently, the system accounts for a total of 14 biological classes as follows:

- gene
  - metabolic gene
  - regulatory gene
- protein
  - transcription factor
  - enzyme
- pathway
- reaction
- compound
- organism
- DNA
- RNA
- physiological state
- laboratory technique

This schema allows the user to identify molecular entities that may describe different levels of biological organisation and thus, lead to a better insight in functional description of cellular processes.

For instance, a physiological state is frequently characterised by particular level of defined biological entities, like compounds catalysed by certain enzymes, which in turn are encoded by the respective genes. Besides common annotation, this schema also supports annotation linking to lexical resources (Figure 2), i.e., it identifies the dictionary entry that triggered each tagging as well as the normalised term (the "concept label" that gathers together known variants and synonyms of a given term).

The ability to use other annotation schemas is considered a premise of tool interoperability and data re-use. As such, annotation schemas derived from the GENIA ontology (Kim et al., 2003), a formal model of cell signaling reactions in human, or used in challenges such as Biocreative, often referenced by the research community as gold standards, were accounted for. It is possible to choose which schema to use on a given annotation task and also to translate from one schema to another. Additionally, we devise the incorporation of new schemas as long as the user specifies tagging and mapping functions.

Regarding POS, the premise is similar and thus, we chose to incorporate GATE for the development language processing components. GATE provides a reusable design and a set of prefabricated software building blocks (namely tokenizers, sentence splitters and POS taggers) that can be used, extended and customised for specific needs. Also, its component-based model allows for easy coupling and decoupling of the processors, thereby facilitating comparison of alternative configurations or different implementations of the same module (e.g., different parsers). At Figure 2, we illustrate an example of POS tagging output.

Figure 2. Small piece of an annotated document using the default annotation schema and GATE default POS tagging.

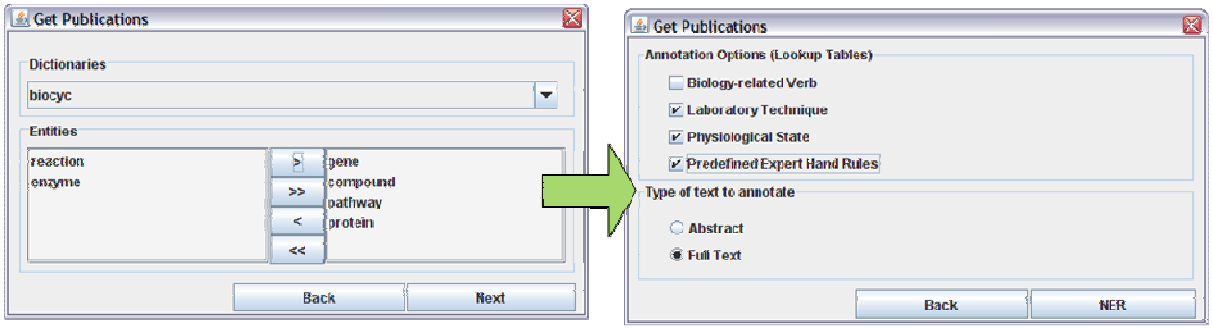


Figure 3. Configuring the automated lexical-based BioNER process.

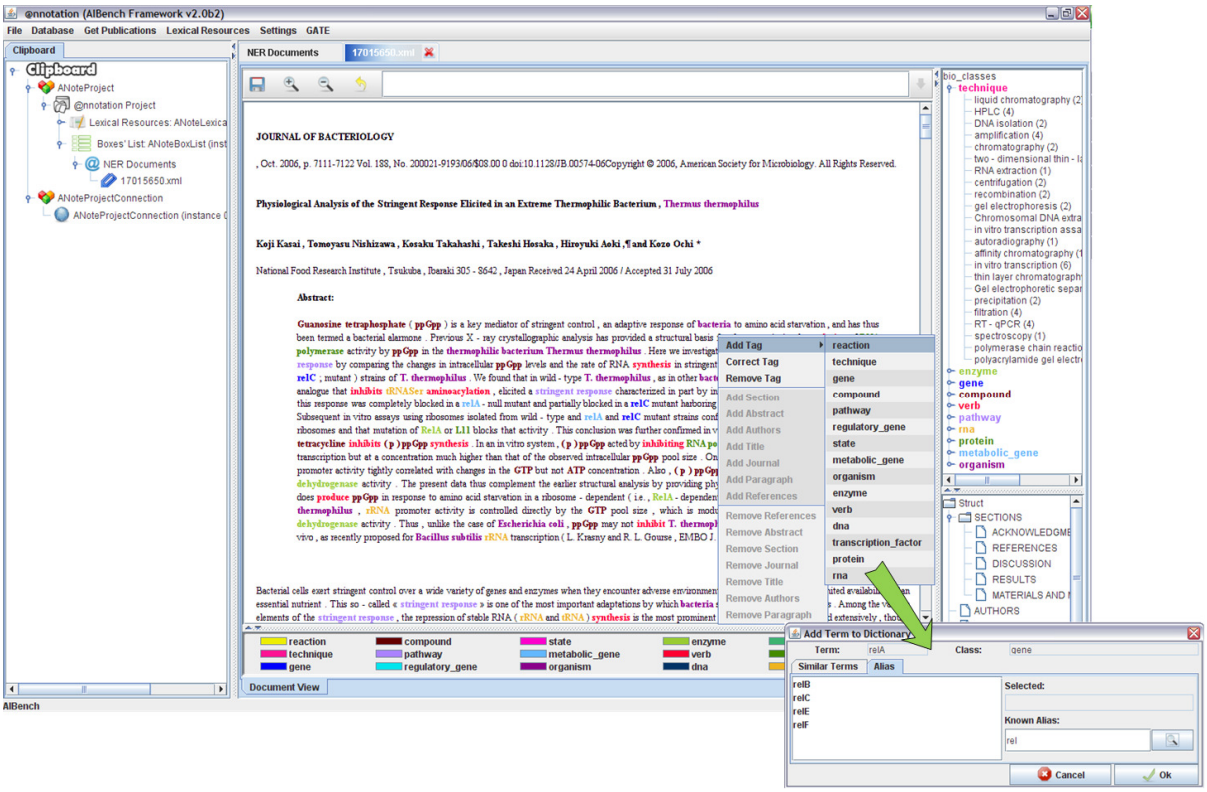


Figure 4. Snapshot of the manual annotation environment.

4.3 Automatic Annotation

The conversion of source formats into plain text is carried out by freeware programs such as Xpdf[10] (Windows or Linux) and pdftotext[11] (Mac OS). The process of XML-oriented document structuring was implemented by the authors using simple pattern matching. Documents (abstracts or full-texts) are submitted to tokenising and stopword removal processes, implemented using Lingua::PT::PLNbase and Lingua::StopWords Perl modules, respectively.

Following the pre-processing step, lexicon-based BioNER is sustained by a specialised rewriting system developed by the authors upon the Text::RewriteRules Perl module. The user specifies the supporting dictionary and the set of biological classes to be annotated (Figure 3). Lookup tables and general templates may also be included. Furthermore, the process can be deployed over abstracts or full-texts.

The system attempts to match terms against dictionary and lookup table contents, checking for different term variants (e.g. hyphen and apostrophe variants) and excluding too short terms (less than 3-character long). Annotation gives preference to longest term matching, tracking up to hepta-grams (i.e. 7-word composition).

Additional patterns account for previously unknown terms and term variants. For example, the template ”([a-z]{3}[A-Z]+\d*)” (a sequence of three lower-case letters followed by an upper-case letter and a sequence of zero or more digits) is used to identify candidate gene names while the categorical nouns ”ase” and ”mRNA” track down possible enzyme and RNA mentions, respectively. Besides class identification, the system also sustains term normalisation, grouping all term variants around a “common name” for visualisation and statistical purposes.

4.4 Manual Annotation

The manual annotation environment accounts for the review of automatic annotations by experts and the enhancement of the lexical resources. Also, manually curated documents are intended to be further used as training corpora to build annotation, classification or other generalised learning models regarding biomedical contents.

Although the actual corpus file with annotation is encoded in XML, the annotators work on

[10] http://www.foolabs.com/xpdf/
[11] http://www.bluem.net/downloads/pdftotext_en/

a CSS-styled view which is much more user-friendly (Figure 4). Furthermore, a query view is used to depict the relation of the annotated terms with dictionary entries.

When the user revises dictionary-based annotation and corrects or adds annotations, the dictionary is updated with such previously unknown or mischaracterised information. Therefore, this process has two major outputs: high-quality annotation and dictionary enrichment. The latter is a classical example of a process of learning by experience that accounts for well-known biological issues such as term novelty, term synonymy and term homonymy. Term novelty and the association of synonyms are far from being adequately tackled as they will depend on expert's knowledge, which is limited and often outdated just like dictionaries. However, the disambiguation of distinct mentions using the same term (e.g. same gene, protein and RNA name) is a classical example where manual curation is invaluable.

Also, users may cooperate on curation tasks, sharing locally processed documents and taking advantage of dictionaries that have been refined by other users.

5 Conclusions

The need for user-friendly and interoperable semantic annotation tools is indisputable in BioTM. Research benefits greatly from the reuse of data (such as annotated corpora) and the capacity to interchange tools (namely POS and semantic taggers). However, this is only possible if tools are devised for this purpose, i.e., if they account for general annotation as well as annotation interchange and if processing tools are prepared to account for distinct annotation schemas. On the other hand, annotation is a laborious and time-consuming task that requires from the curators both expertise on the subjects and critical judgment. In this sense, it is very important that annotation tools take advantage of data mining models and available knowledge resources, minimising manual curation efforts, and at the same time, provide for a user-friendly environment.

In this work, a contribution to these issues is provided, with the development of a novel interoperable and user-friendly software tool for biomedical annotation. Its primary contributions are as follows: the ability to process abstract and full-texts interchangeably; a basic semantic annotation schema encompassing embracing all major biomedical entity classes (genes, proteins, compounds and organisms) and some uncommon, although valuable classes (laboratory techniques and physiological states); the ability to use standard annotation schemas such as GENIA; a preprocessing module capable of converting documents from common file formats (such as PDF and HTML) to plain text and then, tokenise and remove stopword from such texts; a lexicon-based biomedical annotator for annotating biomedical texts which allows the construction of customised dictionaries as well as user-defined rules and lookup tables; a user-friendly annotation view that allows the user to verify and correct annotations and refine dictionary contents.

The tool can be used as a stand-alone environment or it can be integrated in a more comprehensive BioTM framework. Currently, it is incorporated in the @Note Biomedical Text Mining workbench[12] (Lourenço et al., 2008). Here, tool interoperability enables automatic information retrieval (PubMed keyword-based query and document retrieval from open-access and subscribed web-accessible journals) as well as mining experiments (using annotated corpora to construct BioNER models).

Future work includes the enhancement of annotation skills based on curator suggestions and the implementation of several measures to minimize discrepancies of inter-annotation and maintain the quality of annotation. Semantic type checking and detection of anomalies in the resulting annotations are devised as the first steps.

The tools are freely available from http://sysbio.di.uminho.pt/anote.php.

Acknowledgments

This work is partly funded by the research projects recSysBio (ref. POCI/BIO/60139/2004) and MOBioPro (ref. POSC/EIA/59899/2004) financed by the Portuguese Fundação para a Ciência e Tecnologia. The work of Sónia Carneiro is supported by a PhD grant from the Fundação para a Ciência e Tecnologia (ref. SFRH/BD/22863/2005).

References

S. Ananiadou, D. B. Kell and J. I. Tsujii (2006). Text mining and its potential applications in systems biology. *Trends Biotechnol., 24,* 571-579.

[12] http://sysbio.di.uminho.pt/anote.php

H. Cunningham, D. Maynard, K. Bontcheva and V. Tablan (2002). GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02).

R. A. A. Erhardt, R. Schneider and C. Blaschke (2006). Status of text-mining techniques applied to biomedical text. *Drug Discovery Today, 11,* 315-325.

K. Fundel and R. Zimmer (2006). Gene and protein nomenclature in public databases. *BMC Bioinformatics, 7*.

D. Hanisch, K. Fundel, H. T. Mevissen, R. Zimmer and J. Fluck (2005). ProMiner: rule-based protein and gene entity recognition. *BMC Bioinformatics, 6*.

Z. Z. Hu, M. Narayanaswamy, K. E. Ravikumar, K. Vijay-Shanker and C. H. Wu (2005). Literature mining and database annotation of protein phosphorylation using a rule-based system. *Bioinformatics, 21,* 2759-2765.

L. J. Jensen, J. Saric and P. Bork (2006). Literature mining for the biologist: from information retrieval to biological discovery. *Nature Reviews Genetics, 7,* 119-129.

J. D. Kim, T. Ohta, Y. Tateisi and J. Tsujii (2003). GENIA corpus--semantically annotated corpus for bio-textmining. *Bioinformatics, 19 Suppl 1,* i180-i182.

J. D. Kim, T. Ohta and J. Tsujii (2008). Corpus annotation for mining biomedical events from literature. *BMC Bioinformatics, 9*.

Z. Kou, W. W. Cohen and R. F. Murphy (2005). High-recall protein entity recognition using a dictionary. *Bioinformatics, 21 Suppl 1,* i266-i273.

Kulick S, Bies A, Liberman M, Mandel M, McDonald R, Palmer M, Schein A and Ungar L (2004). Integrated Annotation for Biomedical Information Extraction. NAACL/HLT Workshop on Linking Biological Literature,Ontologies and Databases: Tools for Users (pp. 61-68).

A. Lourenço, R. Carreira, S. Carneiro, P. Maia, D. Glez-Peña, F. Fdez-Riverola, E. C. Ferreira, I. Rocha and M. Rocha (2008). @Note: a flexible and extensible workbench for Biomedical Text Mining. *Submitted to BMC Bioinformatics*.

T. Morton and J. LaCivita (2003). WordFreak: an open tool for linguistic annotation. Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: Demonstrations (pp. 17-18). NJ, USA: Association for Computational Linguistics Morristown.

S. Mukherjea, L. V. Subramaniam, G. Chanda, S. Sankararaman, R. Kothari, V. Batra, D. Bhardwaj and B. Srivastava (2004). Enhancing a biomedical information extraction system with dictionary mining and context disambiguation. *Ibm Journal of Research and Development, 48,* 693-701.

H. M. Muller, E. E. Kenny and P. W. Sternberg (2004). Textpresso: An ontology-based information retrieval and extraction system for biological literature. *Plos Biology, 2,* 1984-1998.

J. Natarajan, D. Berrar, C. J. Hack and W. Dublitzky (2005). Knowledge discovery in biology and biotechnology texts: A review of techniques, evaluation strategies, and applications. *Critical Reviews in Biotechnology, 25,* 31-52.

N. Okazaki and S. Ananiadou (2006). Building an abbreviation dictionary using a term recognition approach. *Bioinformatics, 22,* 3089-3095.

L. Shi and F. Campagne (2005). Building a protein name dictionary from full text: a machine learning term extraction approach. *BMC Bioinformatics, 6,* 88.

C. J. Sun, Y. Guan, X. L. Wang and L. Lin (2006). Biomedical named entities recognition using conditional random fields model. *Fuzzy Systems and Knowledge Discovery, Proceedings, 4223,* 1279-1288.

L. Tanabe, N. Xie, L. H. Thom, W. Matten and W. J. Wilbur (2005). GENETAG: a tagged corpus for gene/protein named entity recognition. *BMC Bioinformatics, 6*.

L. Yeganova, L. Smith and W. J. Wilbur (2004). Identification of related gene/protein names based on an HMM of name variations. *Computational Biology and Chemistry, 28,* 97-107.

P. Zweigenbaum, D. mner-Fushman, H. Yu and K. B. Cohen (2007). Frontiers of biomedical text mining: current progress. *Briefings in Bioinformatics, 8,* 358-375.

Genic Interaction Extraction by Reasoning on an Ontology

Alain-Pierre Manine, Erick Alphonse
LIPN, Univ. Paris13/CNRS UMR7030
Laboratoire d'Informatique Paris-Nord
Institut Galilée, Université Paris 13
99 ave. Jean-Baptiste Clément
F93430 Villetaneuse
{alainpierre.manine,
erick.alphonse}
@lipn.univ-paris13.fr

Philippe Bessières
MIG, INRA UR1077
Unité Mathématique,
Informatique et Génome
Institut National de la
Recherche Agronomique
F78352 Jouy-en-Josas
philippe.bessieres
@jouy.inra.fr

Abstract

Information Extraction (IE) systems have been proposed in recent years, to extract genic interactions from bibliographical resources. But they are limited to single interaction relations, and have to face a trade-off between recall and precision, by focusing either on specific interactions (for precision), or general and unspecified interactions of biological entities (for recall). Yet, biologists need to process more complex data from literature, in order to study biological pathways, so an ontology is an adequate formal representation to model this sophisticated knowledge. But the tight integration of IE systems and ontologies is still a current research issue, a fortiori with complex ones that go beyond hierarchies. Here, we propose a rich modeling of genic interactions with an ontology, and show how it can be used within an IE system. The ontology is seen as a language specifying a normalized representation of text. IE is performed by first extracting instances from Natural Language Processing (NLP) modules, then deductive inferences on the ontology language are completed. New instances may be infered, bringing together otherwise scattered textual information. We validated our approach on an annotated corpus of gene transcription regulations in *Bacillus subtilis*. We reach a global recall of 89.3% and a precision of 89.6%, with high scores for the ten semantic relations defined in the ontology.

1 Introduction

Interactions between genes and proteins were long studied, while most of their biological knowledge is not described in structured formats of genomic databanks, but scattered in scientific articles. For this reason, numerous works in recent years have been carried out to design Information Extraction (IE) systems, which aim at automatically extracting genic interaction networks from bibliography (Blaschke et al., 1999; Craven and Kumlien, 1999; Friedman et al., 2001; Krallinger et al., 2007). Relations between biological entities are multiple (protein and gene regulations, DNA binding, phosphorylation, homology relations, etc.). Nevertheless, most IE systems are limited to extract unique relations, and face a trade-off between recall and precision. Some focus on precision by extracting specific interactions, for instance between proteins (Craven and Kumlien, 1999; Rindflesch et al., 2000; Blaschke et al., 1999; Ono et al., 2001; Saric et al., 2005), whereas other stress on recall using general relations (Nédellec, 2005; Fundel et al., 2007). However, this does not take into account the complexity of the data processed by biologists, such as biological pathways (Oda et al., 2008). Therefore, ontologies are a well-motivated formal representation able to convey this complex knowledge, but their utilization in IE, beyond mere conceptual hierarchies, is still a research issue. In this paper, we introduce a rich modeling of genic interactions, and a way to fully integrate an ontology within an IE platform.

We refer to an ontology as a thesaurus (concept and relation hierarchies), along with a logical theory given as a set of inference rules (see e.g. (Gómez-Pérez, 1999)). The ontology is seen as a specification of a normalized and decontextualized text representation. A NLP pipeline extracts a first set of ontology instances, then deductive inferences on the ontology language are completed, deriving more instances. IE results

are a set of concept instances linked by semantic relations. Using several well-defined relations gives the opportunity to model more accurately biological domains, and inference rules reasoning on the ontology are able to gather information otherwise scattered throughout bibliographical databases, and to discover knowledge not explicitly stated in texts. Inference rules may be crafted by the domain expert as part of the ontology design, or automatically learnt by Machine Learning (ML) techniques. We focus on this latter case which has been well-motivated in the context of IE systems, as a generic component to easily adapt them to new domains. However, as opposed to previous approaches, learning takes place in the ontology language to produce deductive rules which hold in the domain ontology. From a ML point of view, the learner uses the ontology as hypothesis language, and instantiations of ontology as example language.

However, as stated by (Friedman et al., 2002), ontologies are not necessarily useful to IE, in the sense that the granularity of the classes between a conceptual and a sublanguage model may differ. We deal with this problem by introducing, along with the ontology, a lexical layer, i.e. relations and classes in an intermediate level of abstraction between raw text and concept. This is in line with (Cimiano et al., 2007; Brickley and Miles, 2005), who propose a lexicon model to map expressions in natural language to their corresponding ontology structure, although none of them address it in an IE context.

We discuss related works using ontologies and ML techniques to support IE systems in section 2. We present our approach where IE is fully specified through the design of a domain ontology along with its lexical layer in the next section. We describe how ML techniques can be applied on the ontology instantiations from a corpus to learn deductive rules which can infer new instances during the extraction process (section 4). And we validate our architecture by defining an ontology of genes transcription in bacteria, and by learning inference rules to extract genic interactions from a corpus of the LLL05 challenge (section 5), to finally give perspectives of our work.

2 Related works

The unifying purpose of the ontology allows us to integrate several aspects not simultaneously handled in related works. Consider the sentence:

> The degR gene is transcribed by RNA polymerase containing sigma D, and the level of its expression is low in a mecA-deficient mutant. (PMID: 10486575.)

Extracting the interaction-related knowledge involves processes occurring in multiple abstraction levels. The biological entities have to be recognized, and properly represented. Simplest lexical variations are captured by Named Entities Recognition (NER), as extensively discussed in (Tanabe and Wilbur, 2002; Park and Kim, 2006). A term–concept connection is assumed by several systems, which use mere conceptual hierarchies, without relation (Miyao et al., 2006; Nédellec, 2005; Saric et al., 2005). Here, we normalize a term as a subgraph of ontology instances, including domain knowledge: in the example, the term “RNA polymerase containing sigma D” may be represented as a *protein complex* relation between an “RNA polymerase” *enzyme* and a “sigma D” *protein*. All the synonyms have to share the same representation (e.g. “EsigmaD” or “RNA polymerase sigma D”). We emphasize the terminology status: while, in the previous expression, (Nédellec, 2005) only tag the “sigma D” protein and inaccurately regard it as the interacting entity, we normalize the full term (“RNA polymerase containing sigma D”). Furthermore, whereas most terminological works focus on nouns, we handle verbal terms: the terms “transcription by EsigmaD” and “transcribed by EsigmaD” will be identically represented.

(Nédellec, 2005; Saric et al., 2004) use respectively a general “genic interaction” relation, or a very specific one. The ontology allows to define various conceptual relations: a transcription event between EsigmaD and degR, and a more general regulation between the mecA mutant and the degR gene.

Furthermore, we do not only provide rules processing on a syntactico-semantic level (Miyao et al., 2006; Alphonse et al., 2004; Daraselia et al., 2004), but using ontology as our representation language, we can reason at a semantic level (see, for instance, the use of inference rules in OWL (Mcguiness et al., 2004)). In the previous sentence, this allows to deduce that, although the second interaction of the example involves an inhibition (“level of its expression is low”), as a mutant

gene is implied, mecA and degR are linked by an activation. Inferences may be achieved on multiple sentences, inducing knowledge not explicitly present in the text as we will show it in section 5.

Ontologies become preeminent in the IE field, while most authors exploit it punctually. Their structure may offer a basis to craft extraction rules (Saric et al., 2005; Friedman et al., 2001), or a useful disambiguation resource. For instance, (Cimiano, 2003; Gaizauskas et al., 2003) use it to solve coreferences, (Daraselia et al., 2004) selects relevant syntactic graphs from a parser using the structure of an ontology, (Saric et al., 2005) stress the benefit of an ontology to solve some syntactical ambiguities relying on concepts arity. In most IE pipelines, ontology (or conceptual hierarchy) is only applied to enrich the text with semantic categories (Alphonse et al., 2004; Saric et al., 2004). On the contrary, we used the ontology structure throughout the extraction process, as a language to make inferences from text.

ML techniques have often been used to acquire resources for IE systems, like extraction patterns or rules (Huffman, 1996; Riloff, 1996; Craven and Kumlien, 1999; Alphonse et al., 2004), which are related to our approach. However, they are limited to learn from enriched text representation, as opposed to our approach, where learning takes place in the ontology language.

3 Knowledge representation language of an IE system based on an ontology

Historically, following the "General Theory of Terminology" created by Eugene Wüster from the late 1930s, a term is defined as a word or a group of words which correspond to a concept in a pre-existing conceptual model. More recently, some have criticized this doctrine (Rastier, 1995; Bourigault and Jacquemin, 2000): the conceptual model and the terms are not seen anymore as absolute notions, but as the result of an artificial and application-oriented construction process based on a domain-related corpus. In other words, the terminology is not *discovered*, but *constructed*. We follow this latter conception: our conceptual model, the ontology, is seen as a specification of a normalized representation of a text, neglecting some aspects of the discourse, and keeping some other ones. By designing it, we specify an IE system. Hence, the IE process is equivalent to an automatic semantic annotation of text, into which sentence fragments (terms) are normalized as ontology instances.

3.1 Ontology as a representation language

Figure 1 exemplifies a simplified ontology of transcription in bacteria. In this model, the "transcription" of a gene (*"et"*) from a promoter (*"t_from"*) may happen due to the action of a protein (*"t_by"*). Furthermore, a protein results from

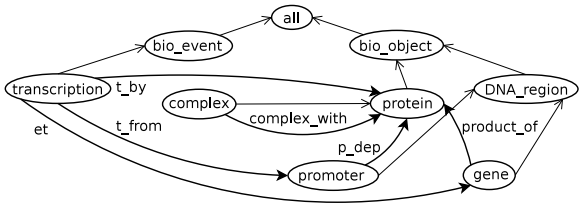


Figure 1: Example of ontology. Labels of "is_a" relations are omitted.

the expression of a gene (*"product_of"*), and a protein complex results from the assembly of several proteins (*"complex_with"*). Figure 2 shows, on an example sentence, the result of the IE system provided as instances of the ontology. Note that, as a normalized representation of the text, not all the meaning is kept: for instance, we do not stress anymore about the "DNA binding" nature of the "GerE" protein; the fact that the transcription happens from "several" promoters is lost. The semantic relations at the bottom of the figure, in plain line, were extracted from text. From the term "transcription from several promoters", a terminological module has extracted instances of "transcription" and "promoter". Then, inferences rules have extracted from text a "t_from" ("transcription from") semantic relation between them. The "p_dep" relation, in bold line in the middle of the figure, is inferred from instances previously extracted from the text, by applying deductive rules on the normalized text representation. This representation fits the specifications

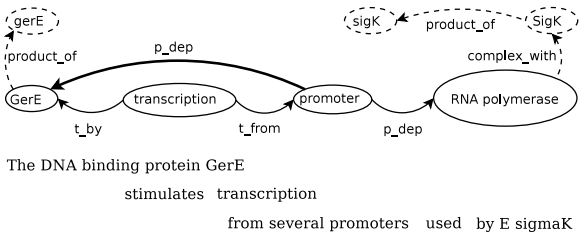


Figure 2: Example of a semantic representation resulting from the IE system.

of the ontology shown in figure 1. Such a rule is the following:

$$\begin{aligned} p\_dep(B, A) \quad \leftarrow \quad & t\_by(C, A), \\ & t\_from(C, B), \\ & protein(A), \\ & promoter(B), \\ & transcription(C). \end{aligned}$$

It means that "if protein A is responsible for a transcription event C from promoter B, then B is dependent on (may be binded by) protein A". Additionally, instances in dotted lines result from domain knowledge: the "GerE" protein is encoded by the "gerE" gene, and the "E sigmaK" protein is a RNA polymerase complexed with the "SigK" protein, itself encoded by the "sigK" gene.

3.2 Features choice for text extraction

Inferences from text require more features. Basically, normalizing a text to a conceptual representation is equivalent to gather multiple lexical forms into a single semantic representation. Hence, the difficulty of the task is related to the complexity of the encountered types of variations. Methods aiming at capturing orthographical and morphological variations are related to Named Entities Recognition (NER), described in (Tanabe and Wilbur, 2002; Park and Kim, 2006). The more complex types of variations are related to relational IE, and processing them involves using NLP tools to enrich the text with syntactic and semantic features. A first set of works builds syntactico-semantic parsers (Friedman et al., 2001; McDonald et al., 2004; Saric et al., 2004; Saric et al., 2005), whereas a second class of systems uses full parsers (Yakushiji et al., 2001; Daraselia et al., 2004; Miyao et al., 2006; Fundel et al., 2007). The latter implies two distinct modules (Yakushiji et al., 2001): a linguistic module, that handles domain-independent structural aspects of the sentence; and an IE module, which is a task-dependent parameter (possibly adapted to the task (Pyysalo et al., 2004)). We follow this general approach which does not involve designing a new syntactico-semantic parser for each new application. This impacts the design of the lexical layer we describe in the next section.

3.3 Lexical layer

We introduce a lexical layer along with the ontology, in which we define relevant semantic features. In figure 3, the concept of "regulation" (and in the example, its instance "stimulate"), and the concept of "dependence" (and its instance "use"), are obviously required. Inference rules do not only need semantic features, but also syntactic ones. To specify them, we introduce syntactico-semantic classes and relations in the lexical layer. Following our conception about ontologies, these classes and relations will define normalizations of text in intermediate states of abstraction, between raw text and conceptual level. They are specified in the ontology shown in figure 4, and will be instantiated by a parser and a terminological module. The layer also allows to introduce classes

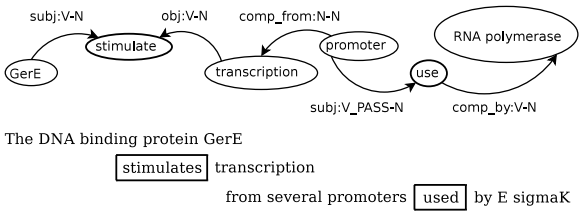


Figure 3: Exemple of a text representation

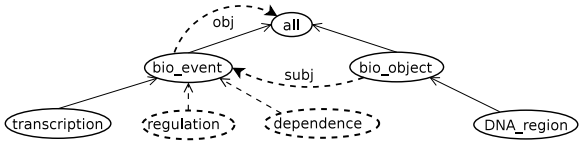


Figure 4: Sample of the lexical layer (elements in dotted line) along with the domain ontology.

which may be semantically irrelevant from a domain ontology point of view but factorize concepts that share common properties, and thus, factorize together otherwise multiple inference rules. This is exemplified in figure 5, which shows the definition of a "biological actor" (bio_actor) class, where a "gene", a "protein" and a "gene family" share common syntactical contexts in biological articles. Figure 3 illustrates a final representation combining semantic features (a protein instance "GerE"), and syntactic ones (a subject "subj:V-N" relation between "GerE" and "stimulate", an

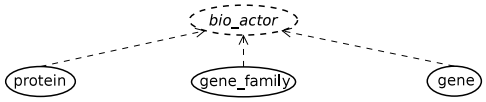


Figure 5: Definition of a syntactico-semantic feature (dotted line) in the ontology.

instance of the "regulation" concept).

4 Acquisition of inference rules

As opposed to previous approaches (see section 2), learning takes place in the ontology language to produce deductive rules which hold in the domain ontology and in the lexical layer. A domain expert has to provide learning examples defined as instantiations of the ontology. He creates instances of concepts and relations of the ontology from a corpus, some instances being output by NLP modules. Target relations are specified to be logically implied by the inference rules. Figure 6 exemplifies such annotation, the dashed lines corresponding to relations to learn.

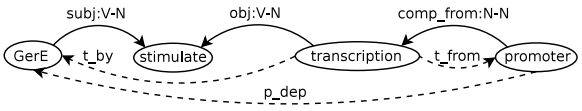


Figure 6: Learning example provided by a semantic annotation.

Learning from such a relational language is known as Inductive Logic Programming (ILP) (Muggleton and Raedt, 1994), where the hypothesis and the example languages are subsets of first-order logic. Most learners handle learning in Datalog which is expressive enough for the task. In Datalog, examples are represented as closed Horn clauses, where the head of the clause is the target relation to learn. For instance, the example of the "t_by" relation in figure 6 will be equivalently represented as the following (relation names have been shorten for presentation):

$$\begin{aligned} t\_by(id1, id2) \quad \leftarrow \quad & subj(id2, id3), obj(id1, id3), \\ & tra(id1, transcription), \\ & pro(id2, "GerE"), \\ & reg(id3, stimulate). \end{aligned}$$

As several relations have to be learnt, learning is set into the multi-class setting where each target relation is learnt in turn, using the other ones as negative examples. Note that all the ontological knowledge is given as background knowledge to the ILP algorithm, like the generalisation relation between concepts. For instance, specifying that a protein complex is a protein, and a protein or a RNA are a gene product, will be represented by a clausal theory:

$$protein(A) \leftarrow protein\_complex(A).$$
$$gene\_product(A) \leftarrow protein(A).$$
$$gene\_product(A) \leftarrow rna(A).$$

Processing an example involving a protein complex or a RNA, the learning algorithm now have the opportunity to choose the most relevant generality level (e.g. "protein complex", "protein" or "gene product") to learn the rules.

5 Results

We validate our architecture by designing an ontology of transcription in bacteria, used to learn inference rules from a *Bacillus subtilis* corpus.

5.1 Ontology encoding biological knowledge

The ontology includes some forty concepts, mainly about biological objects (gene, promoter, binding site, RNA, operon, protein, protein complex, gene and protein families, etc.), and biological events (transcription, expression, regulation, binding, etc.). In the following, we will focus on the ten relations of the ontology.

We defined ten relations: a general interaction relation ("i"), and nine relations specific to some aspects of the transcription (binding, regulons and promoters). Table 1 lists the set of relation names with an example of term. For instance, the third line in the table states that, in the sentence "GerE

Name	Example of related term
p_dep	*sigmaA* recognizes **promoter elements**
p_of	the *araE* **promoter**
b_to	**GerE** binds near the sigK *transcriptional start site*
s_of	*-35 sequence* of the **promoter**
rm	*yvyD* is a member of sigmaB **regulon**
r_dep	*sigmaB* **regulon**
t_from	**transcription** from the Spo0A-dependent *promoter*
t_by	**transcription** by final *sigma(A)-RNA polymerase*
et	**expression** of *yvyD*
i	**KinC** was responsible for Spo0A~P *production*

Table 1: List of relations defined in the ontology, and the corresponding examples of term. Arguments of the relation are shown in italic and bold fonts. The relations are: promoter dependence (p_dep), promoter of (p_of), bind to (b_to), site of (s_of), regulon member (rm), regulon dependence (r_dep), transcription from (t_from), transcription by (t_by), event target (et). "i" is a general interaction relation.

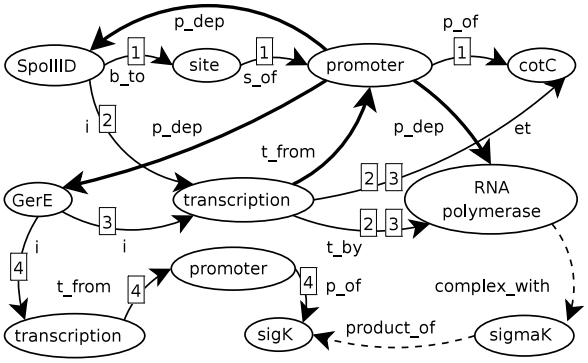


Figure 7: Extracted network from: (1) SpoIIID binds strongly to two sites in the cotC promoter region; (2) SpoIIID represses cotC transcription by sigma(K) RNA polymerase; (3) Transcription of cotC by sigmaK RNA polymerase is activated by GerE; (4) GerE represses transcription from the sigK promoter. Dashed lines represent domain knowledge relations, and bold lines infered ones.

binds near the sigK transcriptional start site", the protein "GerE" (in bold font) binds to (b_to) the site "transcriptional start site" (in italics).

Using an ontology including inference rules, to describe some aspects of the transcription, allows to model biological knowledge more accurately. This is exemplified in figure 7, which shows the instances extracted from four sentences. From the first sentence, inference rules provide the following normalization: SpoIIID binds to (b_to) a site of (s_of) the promoter of (p_of) cotC. The well-defined nature of the involved relations allows to deduce that the cotC promoter is dependent (p_dep) of SpoIIID, as the latter binds to one of its sites. Inferences are not restricted to a sentence: for instance, as the sentence 3 asserts that cotC transcription is activated by GerE, it is possible to deduce that it happens from the cotC promoter (t_from). This latter deduction permits to conclude that the cotC promoter is dependent (p_dep) of GerE. Implicit knowledge distributed into two sentences is therefore made explicit. If less descriptive knowledge is needed, it is easy, by defining a general transitive relation, to provide a database with the genic interacting couples (spoIIID,cotC), (gerE,cotC), (gerE,sigK) and (sigK,cotC). Relations between interacting entities and genes are provided by domain knowledge, as illustrated in the figure with "sigmaK RNA polymerase". The protein complex is known to include protein sigmaK, which is the product of the sigK gene.

5.2 Ontology to learn inference rules

We want to validate the interest of using multiple relations, defined with an ontology, to learn inference rules by ML. In order to test the ontology relevance, we reused the corpus of the LLL05 challenge (Nédellec, 2005), containing 160 sentences, in which we annotated terms, concepts and relations. 541 relations were labeled. Output of NLP tools is complex and heavily noisy, making errors difficult to trace. Thus, to focus exclusively on the rules acquisition task, we only chose to allow as parameters the representation choice and the learning algorithm, the remaining having to be constants and as noiseless as possible. Hence, we enriched and manually curated the linguistic annotations of the LLL05 corpus (parse trees, syntactic categories, lemmas). The representation of the examples was defined following the procedure described in 3.3. We introduced syntactic relations between classes, and syntactico-semantic classes, meant for factorizing entities which may share the same syntactical context: namely, gene and protein, gene family and protein family, transcription and expression events. Eventually, the annotated corpus was used to produce the learning set. To help learning, we added a class of non-interacting biological entities which was generated using the closed-world assumption. We applied the multi-class ILP learner PROPAL (Alphonse and Rouveirol, 2006) to acquire a set of rules for each relation; the non-interacting class was used as negative examples each time but was not learnt. Currently, we only automatically acquire rules involving syntactico-semantic attributes. We will remove this limitation by stratification learning. We provided PROPAL with 541 examples from ten classes, and 10155 from the non-interacting class, and used ten-fold cross-validation, averaged ten times, to evaluate recall and precision of the extraction process. The results are shown in table 2.

As expected, the more specific relations (et, r_dep, rm), assumed to have little lexical variability, are rather trivial to learn, and reach especially high scores. On the contrary, more general ones (i, t_by), exhibiting greater variability, are noticeably harder to learn. We also experiment the two-class case, merging the ten conceptual relations into a positive label, and as shown in table 3, we obtain good recall and precision. Scores are much better than in prelimi-

Relation	Recall	Prec.	Numb.
i	76.4	73.5	161
rm	90.0	90.0	17
r_dep	95.0	100.0	12
b_to	75.0	90.0	14
p_dep	91.5	94.3	47
p_of	87.5	85.2	39
s_of	61.7	80.7	21
et	95.8	99.4	168
t_from	85.0	96.7	18
t_by	65.5	82.6	44

Table 2: Multi-class learning results, for ten fold cross validation averaged ten times, with Recall and Precision in %, and the Number of examples by relation.

nary experiments implying the unique and general "genic interaction" relation from the LLL05 challenge. This corroborates the benefit of using multiple specific relations to model biological knowledge, which involves less complex rules. For instance, in the unique "genic interaction" relation case, the sentences "sigma(H)-dependent expression of spo0A" and "sigma(K)-dependent cwlH gene" would need two rules to be matched (typically, patterns like "A-dependent expression of B" and "A-dependent B"); however, in the multiple relation case, the first sentence would be matched by the patterns "A-dependent B" ("i" relation) and "B of C" ("et" relation), and the second sentence by "A-dependent B" ("i" relation). Thus, in the second case, the "i" rule matches two sentences, where two "genic interaction" rules were needed. By allowing more general rules, the ontology-based approach decreases the required number of examples to be used by the ML algorithm, improving its results.

6 Conclusion and Perspectives

Ontology is a well-motivated formalism to model biological knowledge, and we showed how a domain ontology allows access to knowledge, beyond the capability of current IE systems. However, complex ontologies are not yet fully exploitable in IE systems, which often limit their use to enrich textual data. In this paper, we proposed an original integration of ontology into IE systems. We use the ontology as a language to make inferences on the semantic level, as well as the syntactico-semantic level, thanks to the addition of a lexical layer. IE is performed by first extracting a set of instances from NLP modules,

Recall (%)	Prec. (%)
89.3	89.6

Table 3: Results for two classes learning, using ten fold cross validation averaged ten times.

then deductive inferences on the ontology language are performed, to complete the extraction process. We validated the approach by designing an ontology of genic interactions, and used Machine Learning techniques to learn inference rules from a *Bacillus subtilis* corpus. From a ML point of view, we use the ontology as hypothesis language, and instances of this ontology as example language.

We are currently extending the ontology to handle more phenomenons, especially inhibition/activation distinction, and non-genic actors (e.g. environmental factors). Also, from an operational perspective, we aim at fully automatizing our system by linking the lexical layer to an available NLP pipeline. Notably, as the representation choice is a crucial step in ML, its declarative definition through the ontology is a significant contribution. We then plan to work on text representation, through a comparative study of several lexical layers.

Acknowledgements

We thanks Thierry Poibeau for his useful comments and suggestions on the manuscript. We are grateful to INRA for awarding a Doctoral and Postdoctoral Fellowship to Alain-Pierre Manine.

References

E. Alphonse and C. Rouveirol. 2006. Extension of the top-down data-driven strategy to ILP. In *Proc. Conf. Inductive Logic Programming*. pages 49–63.

E. Alphonse, S. Aubin, Ph. Bessières, G. Bisson, T. Hamon, S. Laguarigue, A. Nazarenko, A.-P. Manine, C. Nédellec, M. Ould Abdel Vetah, T. Poibeau, and D. Weissenbacher. 2004. Event-based information extraction for the biomedical domain: the Caderige project. In *Proc. Intl. Joint Workshop NLP in Biomedicine and its Applications*, pages 43–49.

C. Blaschke, M.A. Andrade, C. Ouzounis, and A. Valencia. 1999. Automatic extraction of biological information from scientific text: Protein-Protein interactions. In *Proc. Seventh Intl. Conf. Intelligent Systems for Molecular Biology*, pages 60–67.

D. Bourigault and C. Jacquemin. 2000. Construction de ressources terminologiques. In J.-M. Pierrel, editor, *Ingénierie des langues*, pages 215–233.

D. Brickley and A. Miles. 2005. SKOS Core Vocabulary Specification. *Technical report, W3C Working Draft*.

P. Cimiano, P. Haase, M. Herold, M. Mantel, and P. Buitelaar. 2007. LexOnto: A model for ontology lexicons for ontology-based NLP. In *Proc. OntoLex07 Workshop*.

P. Cimiano. 2003. Ontology-driven discourse analysis in GenIE. In *Proc. Intl. Conf. Applications of Natural Language to Information Systems, LNI-29*, pages 77–90.

M. Craven and J. Kumlien. 1999. Constructing biological knowledge bases by extracting information from text sources. In *Proc. Intl. Conf. Intelligent Systems for Molecular Biology*, pages 77–86.

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

C. Friedman, P. Kra, H. Yu, M. Krauthammer, and A. Rzhetsky. 2001. GENIES: a natural-language processing system for the extraction of molecular pathways from journal articles. *Bioinformatics*, 17(Suppl. 1):S74–S82.

C. Friedman, P. Kra, and A. Rzhetsky. 2002. Two biomedical sublanguages: a description based on the theories of Zellig Harris. *J. Biomedical Informatics*, 35(4):222–235.

K. Fundel, R. Küffner, and R. Zimmer. 2007. RelEx — relation extraction using dependency parse trees. *Bioinformatics*, 23(3):365–371.

R.J. Gaizauskas, G. Demetriou, P.J. Artymiuk, and P. Willett. 2003. Protein structures and information extraction from biological texts: The PASTA system. *Bioinformatics*, 19(1):135–143.

A. Gómez-Pérez. Ontological engineering: A state of the art. *Expert Update*, 2(3):33–43, 1999.

S.B. Huffman. 1996. Learning Information Extraction patterns from examples. *LNCS*, 1040:246–260.

M. Krallinger, F. Leitner, and A. Valencia. 2007. Assessment of the second BioCreative PPI task: Automatic extraction of protein-protein interactions. In *Proc. Second BioCreative Challenge Evaluation Workshop*, pages 41–54.

D.M. McDonald, H. Chen, H. Su, and B.B. Marshall. 2004. Extracting gene pathway relations using a hybrid grammar: the Arizona Relation Parser. *Bioinformatics*, 20(18):3370–3378.

D. L. McGuinness and F. van Harmelen. 2004. OWL web ontology language overview. *W3C Recommendation*.

Y. Miyao, T. Ohta, K. Masuda, Y. Tsuruoka, K. Yoshida, T. Ninomiya, and J. Tsujii. 2006. Semantic retrieval for the accurate identification of relational concepts in massive textbases. In *Proc. COLING-ACL 2006*, pages 1017–1024.

S. Muggleton and L. De Raedt. 1994. Inductive logic programming: Theory and methods. *J. Logic Programming*, 19,20:629–679.

C. Nédellec. 2005. Learning language in logic — genic interaction extraction challenge. In *Proc. Fourth Learning Language in Logic Workshop*, pages 31–37.

K. Oda, J.-D. Kim, T. Ohta, D. Okanohara, T. Matsuzaki, Y. Tateisi, and J. Tsujii. 2008. New challenges for text mining: mapping between text and manually curated pathways. *BMC Bioinformatics*, 9(Suppl. 3):S5.

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

J.C. Park and J.-J. Kim, 2006. *Text Mining for Biology*, chapter Named Entity Recognition. Artech House Books.

S. Pyysalo, F. Ginter, T. Pahikkala, J. Koivula, J. Boberg, J. Järvinen, and T. Salakoski. 2004. Analysis of link grammar on biomedical dependency corpus targeted at protein-protein interactions. In *Proc. Intl. Joint Workshop NLP in Biomedicine and its Applications*, pages 15–21.

F. Rastier. 1995. Le terme: entre ontologie et linguistique. In *La banque des mots*, pages 35–65. CILF.

E. Riloff. 1996. Automatically generating extraction patterns from untagged text. In *Proc. Natl. Conf. Artificial Intelligence (AAAI)*, pages 1044–1049.

T.C. Rindflesch, L. Tanabe, J.N. Weinstein, and L. Hunter. 2000. EDGAR: extraction of drugs, genes and relations from the biomedical literature. In *Proc. Fifth Pacific Symp. Biocomputing*, pages 517–528.

J. Saric, L.J. Jensen, R. Ouzounova, I. Rojas, and P. Bork. 2004. Large-scale extraction of gene regulation for model organisms in an ontological context. In *Ontology Special of the Third Workshop on Ontology and Genome — Development and Applications of Ontologies on OMICS Research*.

J. Saric, L.J. Jensen, R. Ouzounova, I. Rojas, and P. Bork. 2005. Large-scale extraction of protein/gene relations for model organisms. In *Intl. Symp. Semantic Mining in Biomedicine 2005*.

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

A. Yakushiji, Y. Tateisi, Y. Miyao, and J. Tsujii. 2001. Event extraction from biomedical papers using a full parser. In *Proc. Sixth Pacific Symp. Biocomputing*, pages 408–419.

Combining Multiple Layers of Syntactic Information for Protein-Protein Interaction Extraction

Makoto Miwa[†] **Rune Sætre**[†] **Yusuke Miyao**[†] **Tomoko Ohta**[†] **Jun'ichi Tsujii**[†‡*]
[†]Department of Computer Science, the University of Tokyo, Japan
Bunkyo-ku, Hongo 7-3-1, Tokyo, Japan.
[‡]School of Computer Science, University of Manchester, UK
[*]National Center for Text Mining, UK
{mmiwa,rune.saetre,yusuke,okap,tsujii}@is.s.u-tokyo.ac.jp

Abstract

Protein-protein interaction extraction is a challenging information extraction task in the BioNLP field. Several kernels focusing on a part of syntactic information have been proposed for the task. In this paper, we propose a method to combine multiple layers of syntactic information by using a combination of multiple kernels based on several different parsers. We evaluated the method using support vector machine and achieved an F-score of 62.0% on the AIMed corpus. Further, we analyzed the performance with or without including self-interaction pairs, and found that there is a danger of confusing classifiers and decreasing the performance when treating self-interaction pairs together with real pairs.

1 Introduction

With the growing number of research papers, researchers have difficulty finding the papers that they need. Biomedical relationships in papers help biomedical researchers to find specific papers. Protein-protein interactions (PPI) are one of relations, and automatic extraction methods can help to extract biomedical relationships based on PPI. PPI is also important in biological processes, and finding them automatically can help construct PPI databases that are usually manually constructed, like BIND (Mathivanan et al., 2006). To achieve this, researchers in the BioNLP field have been examining automatic extraction of PPI from research papers.

One major approach for this task is finding a criteria to judge whether a sentence which contains a pair of proteins actually implies interaction of the pair or not. Detection of PPI was initially tackled by using simple methods based on co-occurrences (Blaschke et al., 1999), while more sophisticated NLP techniques have been used later (Bunescu et al., 2005). For example, NLP tools were used to lemmatize surface words and tag them by their parts of speech (POS). Dependency relations in sentences can also be revealed by syntactic parsers. While NLP techniques make this information explicit, appropriate techniques should be applied to use the information collectively for judging the relevance of a sentence for PPI. For this purpose, several kernels have been proposed, including subsequence kernels (Bunescu and Mooney, 2005b), tree kernels (Moschitti, 2006; Sætre et al., 2007), shortest path kernels (Bunescu and Mooney, 2005a), and graph kernels (Airola et al., 2008). Each kernel utilizes a portion of the structures to calculate useful similarity. The kernel cannot retrieve the other important information that may be retrieved by other kernels.

In this paper, we propose a way of combining kernels based on several syntactic parsers for PPI extraction. In order to retrieve the widest range of important information in a given sentence, it is important to extract as much information as possible from the sentence and its parse graphs. Using a support vector machine (SVM) with much useful information from the combination, we achieved an F-score of 62.0% on the AIMed corpus.

We also analyzed the performance changes with or without self-interaction pairs (self-pairs). From this analysis, we found that the self-pairs can have confuse the classifiers and decrease the performances. To avoid this and make the performance better, predicting the self-pairs and the binary-interaction pairs (binary-pairs) indepen-

XPG$_{p1}$ protein interacts with multiple subunits of **TFIIH**$_{prot}$ and with **CSB**$_{p2}$ protein.

Figure 1: A sentence including an interacting protein pair (p1-p2). (AIMed PMID 8652557, sentence 9, pair 3)

before	–
middle	PROT, and, interact, multiple, of, protein, subunit, with
after	protein

Table 1: BOW features

dently could be a better option.

2 The Combination of Kernels Based on Syntactic Parsers

In recent years, parsing technologies have improved rapidly and many different types of parsers have been proposed. Some of them are retrained using biomedical corpora and adapted to biomedical texts (Miyao et al., 2008). Kernel methods applicable to structured data have also been researched. Several kernels are adapted to the parsers' outputs and applied to PPI extraction. The parsers produce different types of structures providing different information regarding the target sentence. Each kernel uses different aspects to extract and utilize some portion of information from the outputs of their parsers. We combine the kernels for utilizing the multiple layers of information that the parsers and the kernels extracted. To realize our method, the PPI extraction method (Sætre et al., 2007; Miyao et al., 2008) is extended. We adopt two types of parsers and three kernels.

2.1 Syntactic Parsers

There are many types of parsers that output different layers of syntactic structures. The structures have different types of useful information. We focus on two types of parsers.

2.1.1 Dependency Parser

The task of a dependency parser is to take a sentence as a sequence of words, and to construct a dependency tree consisting of dependency links between words. Figure 2 is a parse tree produced by a dependency parser.

2.1.2 Deep Parser

A deep parser takes a sentence as a sequence of words like a dependency parser, and constructs graph structures that represent theory-specific syntactic/semantic relations among words. A predicate argument structure (PAS) is often used to represent the semantic structure. It is different from the dependency parser, because it also treats deeper relations and may include reentrant structures. Figure 3 is a parse graph produced by a deep parser.

2.2 Kernels

Syntactic parsers produce useful parse trees or graphs, but the extraction of information from these structures is an open problem. Several kernels are proposed to extract useful information from such structures. Words are also useful features, and several kernels are proposed to treat the combination of the words. We use the following kernels.

2.2.1 Bag-of-words (BOW) kernel

A bag-of-words kernel takes two unordered sets of words as feature vectors, and calculates their similarity. As input, three feature vectors are used. The vectors contain the lemma forms of the words before, inside of, and after the pair (Sætre et al., 2007). The lemmas in the vectors are limited to the top 1,000 most frequent lemmas. Table 1 shows BOW features of the sentence in Figure 1. Polynomial kernels are applied to each feature vector, and their outputs are summed up as the output of the kernel.

2.2.2 Subset tree kernel

A subset tree kernel (Moschitti, 2006) calculates the similarity between two input trees by counting their common subtrees. Subset tree kernels are applied to the shortest path between pairs from a parse tree. The shortest path is calculated including reverse relations to preserve the direction of the parse tree relations. The predicate information in PAS from the deep parser, which was unused in previous works (Sætre et al., 2007; Miyao and Tsujii, 2008), is used to represent the dependency types. An example of shortest path features can be found in Figure 4.

2.2.3 Graph Kernel

A graph kernel (Gärtner et al., 2003; Airola et al., 2008) calculates the similarity between two

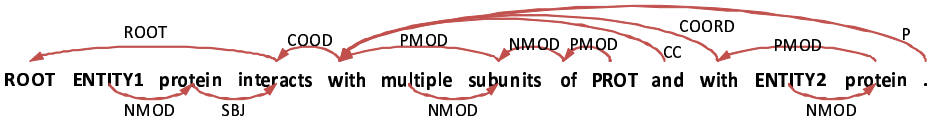


Figure 2: An output example produced by a dependency parser, Kenji and Tsujii's parser. (CoNLL-X dependency tree)

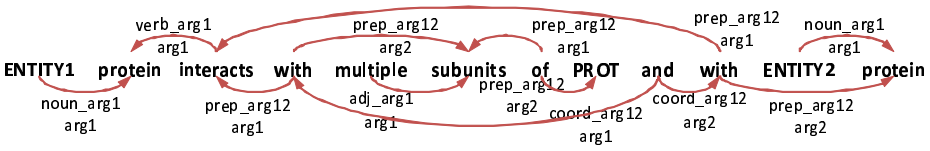


Figure 3: An output example produced by a deep parser, Enju. (Predicate argument structure)

input graphs by comparing the relations between common vertices. This graph kernel is called an all-dependency-paths graph kernel. The weights of the relations are calculated using all walks between each pair of vertices.

The graph consists of two directed subgraphs, a parse graph, and a graph representing a linear order of words. For a vertex in the first subgraph, the dependency and the lemma with POS in the parse graph are used. The dependency and lemma in the shortest path are distinguished from the others. For the PAS structure, the shortest path is calculated by using constituents, and the words in the constituents are distinguished. A vertex in the second subgraph is labeled with the lemma, the POS, and the place information. The place is separated into three by the target pair like BOW features (before, middle, after). Figure 5 is an example of subgraphs made from the parse tree in Figure 2.

For the calculation, two types of matrices are used: a label matrix $\mathbf{L}$, and an edge matrix $\mathbf{A}$. The label matrix is a $N \times L$ matrix, where N is the number of vertices, and L is the number of labels. It represents the correspondence between labels and vertices. $\mathbf{L}_{ij}$ is 1 if the i-th vertex corresponds to the j-th label, and 0 if otherwise. The edge matrix is a $N \times N$ matrix, and represents the relation between pairs of vertices. $\mathbf{A}_{ij}$ is a weight w_{ij} if the i-th vertex is connected to the j-th vertex, and 0 if otherwise. The weight is a predefined constant and the setting is found in the caption of Figure 5. Using the Neumann Series, a graph matrix $\mathbf{G}$ is calculated as:

$$\mathbf{G} = \mathbf{L}^{\mathbf{T}} \sum_{n=1}^{\infty} \mathbf{A}^n \mathbf{L} = \mathbf{L}^{\mathbf{T}}((\mathbf{I} - \mathbf{A})^{-1} - \mathbf{I})\mathbf{L}. \quad (1)$$

This matrix sums up the weights of all the walks between a pair of vertices, so as a result, each entry represents the strength of the relation between a pairs of vertices. Using these two graph matrices, the graph kernel k is defined as:

$$k(\mathbf{G}, \mathbf{G}') = \sum_{i=1}^{L} \sum_{j=1}^{L} \mathbf{G}_{ij} \mathbf{G'}_{ij}. \quad (2)$$

This kernel sums up the products of the common relations' weights.

For fast calculation and performance, the graph kernels of two subgraphs are calculated separately and the normalized outputs of the graph kernels are summed up. In the evaluation, we calculated the matrices of the weights beforehand, and entered the sparse feature vector of the weights into a linear SVM.

2.3 Combination of Kernels

The parsers treat different layers of relations. The dependency parsers ignore some deep information, and conversely, the deep parsers do not output certain shallow relations. Every kernel has different aspects, and has different advantages and disadvantages. The BOW kernels can combine the words easily, but they ignore the internal word order and word relations. The subset tree kernels can calculate the similarity of two shortest paths, but they ignore the words, the paths outside of the shortest path, and cycles in the parsed graphs. The graph kernels can treat the parser's output and word features at the same time. However, they cannot treat them properly without tuning kernel parameters. They may also miss some distant words, and similarities of multiple paths.

(*DEPENDENCY* (NMOD (ENTITY1 protein)) (SBJ (protein interact)) (rCOOD (interact with)) (rCOORD (with with)) (rPMOD (with protein)) (rNMOD (protein ENTITY2)))
(*DEEP* (noun_arg1_arg1 (ENTITY1 protein)) (rverb_arg1_arg1 (protein interact)) (rprep_arg12_arg1 (interact with)) (prep_arg12_arg2 (with protein)) (rnoun_arg1_arg1 (protein ENTITY2)))

Figure 4: Shortest path features

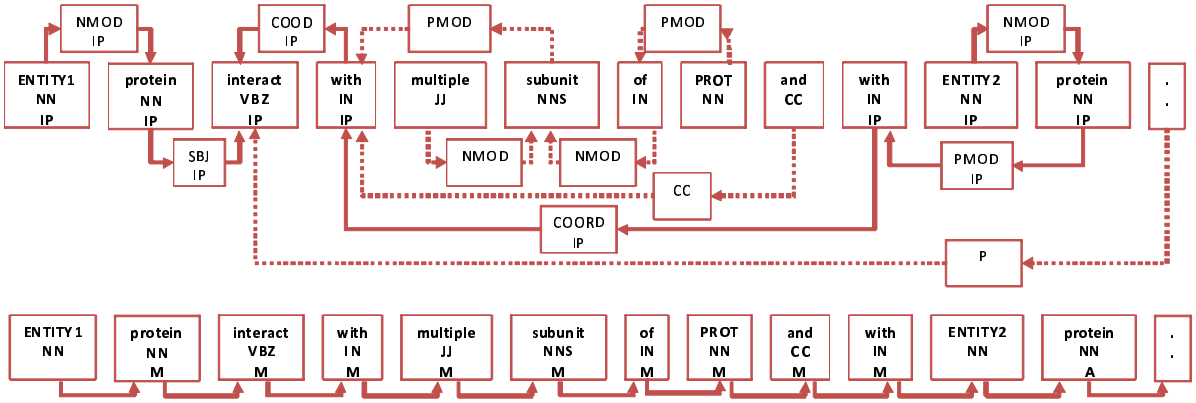


Figure 5: Two directed subgraphs features. One subgraph is made from a parse graph. As a label of a vertex in the subgraph, the relation with the place (IP:In shortest Path) and the lemma with the POS and the place are used. The other graph represents the linear order of words. As a label of a vertex in the subgraph, the lemma with the POS and the place (B:Before, M:Middle, A:After) is used. Weights are assigned to the edges in the subgraphs (Airola et al., 2008), 0.9 for the edges in the shortest paths and the second graph (represented with full lines), and 0.3 for the other edges (represented with dotted line).

The kernels calculate the similarity with different aspects between the two sentences. Combining the similarities can reduce the danger of missing important features, and can produce a new useful similarity measure. To realize the combination of the different types of kernels based on different parse structures, we sum up the normalized output of several kernels k_{ij} as:

$$k(X, X') = \sum_{i}^{K} \sum_{j}^{P} k_{ij}(X, X'), \qquad (3)$$

where K represents the number of types of kernels and P represents the number of parsers. This is a very simple combination, but the resulting kernel function contains all of the kernels' information.

3 Experiments

3.1 Experimental Settings

In the following experiments, we used AIMed (Bunescu and Mooney, 2004)[1], which is a major corpus for the evaluation of PPI extraction methods. We pre-processed AIMed for the named-entity tokenization in the following way. First, we converted spaces in protein names into "_", to group named entities. Then, we put a space between the end of a protein tag and the beginning of another protein tag when they were contiguous. Finally, we converted " to " or " according to their portions. AIMed consists of 225 abstracts (1970 sentences), and we extracted 5,648 binary-pairs including 1,005 positive pairs and 4,233 self-pairs including only other 54 positive pairs. Two examples of self-pairs are shown in Figure 6. Because of the pre-processing, the number of extracted pairs differs from other reported PPI extraction methods. Pre-processing can affect the performance and make it difficult to compare the result. The protein names in a sentence were converted to ENTITY1, ENTITY2, or PROT according to which pair was being processed. Examples are

Suramin induced high-affinity trimerization of *C8self-pair* (Kd = 0.10 microM at 20 degrees C) and dimerization of *C9self-pair* (Kd = 0.86 microM at 20 degrees C) .

Figure 6: Self-interaction pair examples. Each protein interacts with itself. (AIMed PMID 10346902, sentence 6, pair 4 and 5)

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

shown in Figures 2 and 3.

Our system is based on the AKANE++ PPI system[2] (Sætre et al., 2007). We utilized Sagae and Tsujii's dependency parser[3] (Sagae and Tsujii, 2007) as the dependency parser, and the Enju parser[4] (Miyao and Tsujii, 2008) as the deep parser. Both parsers were retrained using the GENIA Treebank corpus[5] (Kim et al., 2003). As word features, we used the lemma information from the Enju parser, which originally output from the GENIA Tagger[6]. We used SVM for the evaluation. The performance was measured in an abstract-wise 10-fold cross validation (CV), and a one-answer-per-occurrence criterion, which were used for the evaluation of other PPI extraction methods before (Giuliano et al., 2006). We controlled the separating hyperplane of the SVM by varying the threshold and calculated the average of the results for each threshold. We fixed the other parameters, and we set the regularization parameter C to 1. We report the best f-value for each SVM in the following tables.

3.2 Effects of Self-pairs

We extracted 54 self-pairs from AIMed. The number of the self-pairs is much smaller than the number of the binary-pairs. Most previous results were obtained without the self-pairs. We evaluated the performance of our method in three different ways:

1. Evaluation without including any self-pairs
2. Evaluation without trying to predict any self-pairs
3. Evaluation with prediction of self-pairs

The first way ignores the self-pairs in prediction and evaluation. The result is shown in Table 2. We also showed the performance of the co-occurrence (or all-true) baseline. The result looks better than others, but it is too optimistic to assume that self-pairs can be classified with the same performance as the binary-pairs at the same threshold. However, this way is useful for comparing our method with other reported methods.

[2]http://www-tsujii.is.s.u-tokyo.ac.jp/~satre/akane/
[3]http://www.cs.cmu.edu/~sagae/parser/
[4]http://www-tsujii.is.s.u-tokyo.ac.jp/enju/
[5]http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/
[6]http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/tagger/

	K	E	K+E
C (baseline)	30.2	–	–
B	53.9	–	–
T	55.5	54.6	57.0
G	59.2	58.4	60.6
T+B	60.1	59.6	60.4
G+B	60.0	59.5	61.0
T+G	62.1	61.6	62.6
T+G+B	62.6	62.7	64.3

Table 2: F-score without including any self-pairs (K:Sagae and Tsujii's parser, E:Enju, C:Co-occurrence, B:BOW, T:Subset Tree, G:Graph, +:Summation of kernels)

	K	E	K+E
CF (baseline)	30.0	–	–
B	52.6	–	–
T	54.0	53.2	55.7
G	57.8	57.1	59.2
T+B	58.7	58.2	59.0
G+B	58.5	58.1	59.4
T+G	60.4	60.2	61.1
T+G+B	61.1	61.1	62.7

Table 3: F-score without trying to predict any self-pairs (CF:Co-occurrence on binary-pairs and all-false on self-pairs)

	K	E	K+E
CF (baseline)	30.0	–	–
B	51.1	–	–
T	54.0	53.2	55.7
G	58.2	56.7	59.2
T+B	58.0	57.2	58.7
G+B	59.0	57.8	59.1
T+G	60.7	60.5	61.4
T+G+B	60.2	60.1	61.2

Table 4: F-score with prediction of self-pairs

The second way ignores the self-pairs in prediction, but adds the positive self-pairs as false negatives during evaluation. This way is the same as applying an all-false method to the self-pairs. The result is shown in Table 3. The results are always lower than the first one because of the self-pairs. The baseline was also calculated using the all-false method to the self-pairs. The third way considers the self-pairs in prediction and evaluation. The result is shown in Table 4. This is the right way when we want to make the PPI extraction system simple or we place much trust in machine learning methods. The baseline method is the same as the second one.

Some of the results using the graph kernels in the third way performed better than those in the

second way. The graph kernel can acquire the features on the self-pairs, and the graph kernel is one of the hopeful approaches in treating both pairs together. In other cases, however, the self-pairs confused the classifiers and the scores were low especially with the BOW kernels. It should be noted that the best threshold in Table 2 was different from the one in Table 4. This indicates that there is a suited threshold for each type of interaction pair. We can judge whether an interaction pair is a self-pair or not beforehand, and we do not need to treat both types of pairs together. It is better to predict the self-pairs and the binary-pairs independently to ease the task and improve the performance.

3.3 Accuracy Improvements by Combinations

In Table 3, we can find that the performance increased with an increase in information. This shows that both the combination of kernels and the combination of parsers are effective for PPI extraction. Our method performed best when all information was extracted. This is surprising, because AIMed is very small and there are much relevant information between the graph kernel and the other kernels, which can cause over-fitting.

The comparison with the results of related PPI methods is summarized in Table 5. The averages of Precision, Recall, and the F-score were calculated independently. For the fair comparison, we performed a 10-fold CV in each training data to tune the threshold for each fold, the result of which is shown as the F-scores in parentheses in Table 5. The F-scores in parentheses show that our evaluation method overestimated by only 0.8%, so it is not too optimistic . Additionally, we also calculated AUC (area under the ROC [receiver operating characteristic] curve) and standard deviations provided for the F-score and AUC (Airola et al., 2008). The ROC curve is a plot of TPR (true positive rate) vs FPR (false positive rate) for different thresholds. The F-score is the best point from a Precision-Recall (PR) curve (a plot of Precision vs Recall as we vary the threshold) . Because of these different points of view, the best result in AUC differs from the best result in the F-score. Which result is better depends on the given task; we have thus reported both results.

The results cannot be compared directly, because of the differences in data preprocessing, the different number of target protein pairs, and different evaluation methods. We compare our method with other methods based on the evaluation proposed in other PPI papers. We use the F-score for all the comparisons except for the comparison with (Airola et al., 2008), which used AUC for the first time in PPI extraction.

Our method outperformed all the PPI extraction methods evaluated with the abstract-wise 10-fold CV even though some of them ignored the self-pairs in their prediction and evaluation. The result of our method was 0.8% lower than the result of (Miyao et al., 2008) in the same condition, which is K+E and T+B in Table 4. This is because they did not use the predicate information in the PAS structure. On the other hand, this information increased the performance of the graph kernel. We may need to evaluate our method more precisely in order to decide the optimal input structures. Our method is different from (Airola et al., 2008) in that they performed the leave-one-document-out CV on the training data to tune the parameter. We compare their results with our result without including any self-pairs, because they ignored the self-pairs. Our method performed 7.1% better than theirs in F-score and 0.03 better than theirs in AUC. (Giuliano et al., 2006) performed a different evaluation method. As reported in (Airola et al., 2008), its result was an F-score of 52.4%. (Bunescu and Mooney, 2005b), (Erkan et al., 2007), and (Katrenko and Adriaans, 2006) also performed different evaluation methods. Their methods with our evaluation method is expected to give a lower performance (Sætre et al., 2007; Airola et al., 2008).

3.4 Error Analysis

Figures 7 and 8 show some false positive examples and some false negative examples. Some false positives and false negatives were caused by uncertainty and negation of interactions (Pyysalo et al., 2008). The kernels we used may not be able to distinguish these interactions from the others, because they do not extract modal information related to the interactions. The words representing the interactions may not exist in the shortest path in the subset tree kernel, and the information for the interactions may need relations among more than three words which are not retrieved in the graph kernel. Some synonyms and abbreviations

	positive	all binary	P	R	F	σ_F	AUC	σ_{AUC}
without including any self	1,005	5,648	60.4	69.3	64.3 (63.5)	4.3	0.879	0.026
without trying to predict any self	1,059	5,648	60.4	65.6	62.7 (62.0)	3.5	0.834	0.032
with the prediction of self	1,059	5,648	57.8	66.1	61.4 (60.7)	3.9	0.914	0.020
(Miyao et al., 2008)	1,059	5,648	54.9	65.5	59.5			
(Airola et al., 2008)	1,000	5,834	52.9	61.8	56.4	5.0	0.848	0.023
(Sætre et al., 2007)	1,068	5,631	64.3	44.1	52.0			
(Mitsumori et al., 2006)	1,107	5,476	54.2	42.6	47.7			
(Yakushiji et al., 2005)			33.7	33.1	33.4			

Table 5: Comparison with previous results of the PPI extraction methods with the abstract-wise 10-fold CV on the AIMed corpus. (The F-scores in parentheses were obtained using the 10-fold CV in each training data to tune the threshold for each fold.)

Uncertainty Interaction of p85$_{\text{prot}}$ subunit of *PI 3-kinase$_{prot}$* with *insulin$_{prot}$* and IGF-1$_{\text{prot}}$ receptors analysed by using the two-hybrid system .

Synonyms The catalytic domain of activated *collagenase_I$_{prot}$* (MMP-1$_{\text{pair1, pair2}}$) is absolutely required for interaction with its specific inhibitor , *tissue_inhibitor_of_metalloproteinases-1$_{pair1}$* (TIMP-1$_{\text{pair2}}$) .

Other Misclassification A 51-residue region from the conserved C-terminal region of TBP$_{\text{pair12, pair13, pair14}}$, previously shown to be the binding site for the viral activator protein *E1A$_{pair12}$* , interacts with *c-Fos$_{pair13}$* and c-Jun$_{\text{pair14}}$ proteins.

Figure 7: False positive examples. Mis-detected relations are shown in italic.

were also among the false positives. They cannot be distinguished from the true positives because the protein names are hidden. In false negatives, there were interactions that needed more information of the words and the context for the extraction. Some of the interactions may be extracted by using the incidental features in many texts, but other interactions will not be detected by the current sentence-based approach.

4 Conclusion and Future Work

In this paper, we have proposed an approach using a combination of kernels for PPI extraction, which can in turn extract and combine several different layers of information from a sentence and its syntactic structures by using several parsers. Each kernel extracts some information from the sentence with different aspects and loses the other

Need Information of Context We screened proteins for interaction with *presenilin_(_PS_)_1$_{pair1}$*, and cloned the full-length cDNA of human *delta-catenin$_{pair1}$*, which encoded 1225 amino acids.

Need Information of Context We have engineered and purified recombinant *K5$_{pair5}$* head and *DPI$_{pair5}$* tail , and we demonstrate direct interaction in vitro by solution-binding assays and by ligand blot assays .

Negation This demonstrates that the C-terminal hemopexin domain of MMP-1$_{\text{prot}}$, in contrast to the corresponding regions of *gelatinase_A$_{pair4}$* and *gelatinase_B$_{pair5}$* , does not interact with *TIMP-1$_{pair4, pair5}$* .

Other Misclassification Using the cytoplasmic domain of Fas in the yeast two-hybrid system , we have identified a novel interacting protein , *FADD*$_{\text{pair2, }pair3\text{, pair4}}$, which binds Fas$_{\text{pair2}}$ and *Fas$_{pair3}$* - FD5$_{\text{pair4}}$, a mutant of Fas$_{\text{prot}}$ possessing enhanced killing activity , but not the functionally inactive mutants Fas$_{\text{prot}}$ - LPR$_{\text{prot}}$ and Fas$_{\text{prot}}$ - FD8$_{\text{prot}}$.

Figure 8: False negative examples. Undetected interactions are shown in italic.

information in it. The combination of the kernels can gather up all the kernels' information and cover some of the lost information. To show the usefulness of the combination of kernels and parsers in the PPI extraction, we evaluated our method using the AIMed corpus. We achieved an F-score of 62.0% using a SVM. This result is better than the results of all the current state-of-the-art PPI extraction methods.

We also analyzed how the performance changed when including the self-pairs. The re-

sults of the graph kernels showed that it may be a good solution to treat both kinds of pairs together. We should, however, predict the binary-pairs and the self-pairs independently to make the task easier and to improve the performance, because the best threshold for the binary-pairs is different from the one for the self-pairs. When we predicted the binary-pairs and the self-pairs at the same time, the classifiers produced worse results than the results of the classifiers classifying all the self-pairs as false. We need a way to detect the self-pairs better than the all-false method.

There is the possibility of improvement by adding (or replacing) new kernels and parsers. This is to remedy the fact that the parsers we used may miss some information in a sentence and the kernels we used may not extract full information of the parsers' outputs. There may be relevant and redundant information in the combination of all the kernels, which can confuse classifiers. How redundant features affect the performance need to be analyzed before adopting new kernels and parsers for evaluating them correctly.

Acknowledgments

This work was partially supported by Grant-in-Aid for Specially Promoted Research (MEXT, Japan) and Genome Network Project (MEXT, Japan).

References

A. Airola, S. Pyysalo, J. Björne, T. Pahikkala, F. Ginter, and T. Salakoski. 2008. A graph kernel for protein-protein interaction extraction. In *Proceedings of the BioNLP 2008 workshop*, pages 1–9.

C. Blaschke, M. A. Andrade, C. Ouzounis, and A. Valencia. 1999. Automatic extraction of biological information from scientific text: Protein-protein interactions. In *Proceedings of the AAAI Conference on Intelligent Systems in Molecular Biology*, pages 60–67.

R. C. Bunescu and R. J. Mooney. 2004. Collective information extraction with relational markov networks. In *ACL 2004*, pages 439–446.

R. C. Bunescu and R. J. Mooney. 2005a. A shortest path dependency kernel for relation extraction. In *HLT/EMNLP*, pages 724–731, Morristown, NJ, USA. Association for Computational Linguistics.

R. C. Bunescu and R. J. Mooney. 2005b. Subsequence kernels for relation extraction. In *NIPS 2005*.

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

G. Erkan, A. Ozgur, and D. R. Radev. 2007. Semi-supervised classification for extracting protein interaction sentences using dependency parsing. In *EMNLP 2007*.

T. Gärtner, P. A. Flach, and S. Wrobel. 2003. On graph kernels: Hardness results and efficient alternatives. In *Proceedings of the 16th Annual Conference on Computational Learning Theory and the 7th Kernel Workshop*.

C. Giuliano, A. Lavelli, and L. Romano. 2006. Exploiting shallow linguistic information for relation extraction from biomedical literature. In *EACL 2006*.

S. Katrenko and P. Adriaans. 2006. Learning relations from biomedical corpora using dependency trees. In *KDECB*, pages 61–80.

J.-D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii. 2003. GENIA corpus — a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19:i180–i182.

S. Mathivanan, B. Periaswamy, T. Gandhi, K. Kandasamy, S. Suresh, R. Mohmood, Y. L. Ramachandra, and A. Pandey. 2006. An evaluation of human protein-protein interaction data in the public domain. *BMC Bioinformatics*, 7 Suppl 5:S19.

T. Mitsumori, M. Murata, Y. Fukuda, K. Doi, and H. Doi. 2006. Extracting protein-protein interaction information from biomedical text with SVM. *IEICE - Transactions on Information and Systems*, E89-D(8):2464–2466.

Y. Miyao and J. Tsujii. 2008. Feature forest models for probabilistic HPSG parsing. *Computational Linguistics*, 34(1):35–80.

Y. Miyao, R. Sætre, K. Sagae, T. Matsuzaki, and J. Tsujii. 2008. Task-oriented evaluation of syntactic parsers and their representations. In *Proceedings of ACL-08: HLT*, pages 46–54.

A. Moschitti. 2006. Making tree kernels practical for natural language processing. In *EACL 2006*.

S. Pyysalo, A. Airola, J. Heimonen, J. Björne, F. Ginter, and T. Salakoski. 2008. Comparative analysis of five protein-protein interaction corpora. In *BMC Bioinformatics*, volume 9(Suppl 3), page S6.

R. Sætre, K. Sagae, and J. Tsujii. 2007. Syntactic features for protein-protein interaction extraction. In *LBM 2007 short papers*.

K. Sagae and J. Tsujii. 2007. Dependency parsing and domain adaptation with LR models and parser ensembles. In *EMNLP-CoNLL 2007*.

A. Yakushiji, Y. Miyao, Y. Tateisi, and J. Tsujii. 2005. Biomedical information extraction with predicate-argument structure patterns. In *First International Symposium on Semantic Mining in Biomedicine*.

BioLexicon: A Lexical Resource for the Biology Domain

Yutaka Sasaki [1] Simonetta Montemagni [3] Piotr Pezik [4] Dietrich Rebholz-Schuhmann [4]
John McNaught [1, 2] Sophia Ananiadou [1, 2]

[1] School of Computer Science, University of Manchester
[2] National Centre for Text Mining
MIB, 131 Princess Street, Manchester, M1 7DN, United Kingdom
Yutaka.Sasaki@manchester.ac.uk

[3] Istituto di Linguistica Computazionale, CNR, Via Moruzzi 1, 56124 Pisa, Italy

[4] EBI, Wellcome Trust Genome Campus, Cambridge, CB10 1SD, United Kingdom

Abstract

Natural language processing technologies have advanced remarkably in the past two decades. However, biological terminology is a frequent cause of analysis errors when processing literature written in the biology domain. The BOOTStrep BioLexicon is a linguistic resource tailored for the domain to cope with these problems. It contains the following types of entries: (1) a set of terminological verbs; (2) a set of derived forms of the terminological verbs; (3) general English words frequently used in the biology domain; (4) domain terms. This comprehensive coverage of biological terms makes the lexicon a unique linguistic resource within the domain. This paper focuses on the linguistic aspects of the lexicon.

1 Introduction

Over the past twenty years, there have been remarkable advances in natural language processing (NLP) and text mining (TM) technologies. Various practical NLP/TM tools, such as part-of-speech taggers, chunkers, syntactic parsers and named entity recognizers, are now widely available.

However, text in biology exhibits different characteristics from general language documents such as newspaper articles. The biology domain demonstrates strong demands for the results of NLP/TM. However, it is also one of the most challenging domains for text processing (Ananiadou and McNaught, 2006).

Lack of coverage of the following types of terminological information makes NLP/TM tasks in this domain difficult:

- Large-scale domain-specific terminologies
- Domain-specific word usage
- Domain-specific relations between words

Technical terms are a major barrier to bio-text processing. A huge number of biological, chemical and medical terms appear in the literature and new terms are coined every day. Furthermore, there are many spelling and semantic variants of these terms representing the same biomedical entities in different written forms. For example, the BioThesaurus [1] contains more than 15 million gene/protein names, but still it does not cover the wide variety of variants of gene/protein names actually appearing in the literature.

Word usage can be idiosyncratic to the bio-domain as well. For example, *express* often indicates a specific biological process, *gene expression*, and takes as arguments specific types of named entities, such as gene and protein names.

In addition, there are many cases where words are related in a biology-specific manner. For example, the verb *retroregulate* has *retroregulation* as its nominal form and *retroregulatory* as its adjectival form. This extent

[1] http://pir.georgetown.edu/pirwww/iprolink/biothesaurus.shtml

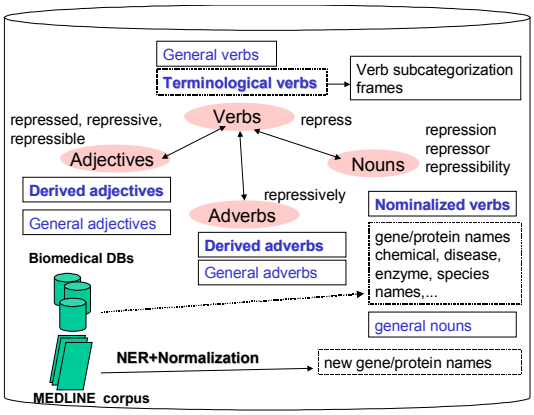


Figure 1 Overview of the Lexicon

of derivational relations between words in the biological domain cannot be fully covered by general English dictionaries and thesauri, *e.g.*, WordNet. To the best of our knowledge, there is no biology-specific lexicon that addresses the above linguistic issues.

2 Overview of the BioLexicon

Figure 1 shows an overview of the BioLexicon. It consists of four part-of-speech categories: verb, noun, adjective, and adverb. Each category accommodates terminological words and general language words. Biology terms, *e.g.*, gene/protein names, are either gathered from existing databases or automatically extracted from text. Other terminological words and their relations are manually curated. Inflections of general words are manually curated based on the MedPost dictionary (Smith *et al.,* 2004).

The database model of the lexicon follows the Lexical Markup Framework (LMF) (Francopoulo *et al*., 2006). The details of the database model were reported in Quochi *et al*. (2008).

3 Biology-relevant terminologies

The terminologies in the lexicon are fivefold: (1) verbs, (2) adjectives, (3) adverbs; (4) terminological nouns, and (5) biomedical terms. (1) – (4) have been manually curated.

(1) Terminological verbs
759 base forms (4,556 inflections) of terminological verbs.

(2) Terminological adjectives
1,258 terminological adjectives.

(3) Terminological adverbs
130 terminological adverbs.

(4) Nominalized verbs
1,771 nominalized verbs.

(5) Biomedical terms
Currently, the BioLexicon contains biomedical terms in the categories of cell (842 entries, 1,400 variants), chemicals (19,637 entries, 106,302 variants), enzymes (4,016 entries, 11,674 variants), diseases (19,457 entries, 33,161 variants), genes and proteins (1,640,608 entries, 3,048,920 variants), gene ontology concepts (25,219 entries, 81,642 variants), molecular role concepts (8,850 entries, 60,408 variants), operons (2,672 entries, 3,145 variants), protein complexes (2,104 entries, 2,647 variants), protein domains (16,940 entries, 33,880 variants), Sequence ontology concepts (1,431 entries, 2,326 variants), species (482,992 entries, 669,481 variants), and transcription factors (160 entries, 795 variants).

In addition to the existing gene/protein names, 70,105 variants of gene/protein names have been newly extracted from 15 million MEDLINE abstracts. Section 5 describes the methods used.

3.1 Terminological verbs

Terminological verbs have been manually curated through examination of biomedical literature. As a result, 759 verbs were selected.

Following the selection of verbs, three types of orthographic variants were added to the lexicon.

- British/American spelling variants
e.g., *acetylise* (British)/*acetylize* (American) or *harbour* (British)/*harbor* (American)

- Hyphenation variants
e.g., *co-activate* and *coactivate*

- Combination of the above two
e.g., co-localise (British), colocalise (British), co-localize (American), colocalize (American)

Inflectional forms are all enumerated in our lexicon. The following verbal inflections have been completely curated.

VV base form
VVD past tense
VVN past participle
VVZ third person singular present
VVG gerund or present participle

The above parts-of-speech follow the Penn Treebank POS tags (Santorini, 1990).

3.2 Derived forms of terminological verbs

Our strategy was to expand the terminology from terminological verbs to derived forms. Three types of derivational relations of the terminological verbs have been introduced. Frequently, nominalized verbs play the same role as verbs. Adjectival and adverbial derived forms may also be used to represent biological events and processes in the same context as their associated verbs. For text mining applications, it is important to cover these possibilities as far as those derivations are linguistically correct.

(1) Nominalization

Nominalized verbs are verbs that are used as nouns. A verb can be nominalized with or without morphological transformation. For example, the nominalized forms of *regulate* are *regulation* and *regulator*. Following Comrie and Thompson (2007), we identified two kinds of nominalization.

(i) Action/state nouns

The noun expresses an action or state of the verb from which it is derived, *e.g.*,

act (v) → action (n),
act (v) → act (n),
act (v) → acting (n).

(ii) Agentive nouns

The noun has an 'agent' role to the verb from which it is derived, *e.g.*,

act (v) → actor (n)

(2) Adjectival derivation

The derivational relation between adjectives and the verbs from which they are derived was manually curated, because there is no dictionary that fully covers adjectival derivations of biological terms. *E.g.*,

act (v) → actable (adj.),
act (v) → active (adj.).

(3) Adverbial derivation

The derivational relation between adverbs and the verbs from which they are derived were also manually curated, *e.g.*,

act (v) → actively (adv.)

3.3 Biomedical terms

Existing biological databases have served as the first source of many nominal types of terms represented in the BioLexicon. Detailed information can be found on the BOOTstrep web site. (Bootstrep, 2008). Such resources are characterized by a high coverage of biological entities and they contain terms annotated with widely recognized and interoperable accession number (*e.g.,* UniProt). On the other hand, some terms imported from existing resources are assigned to concept identifiers in the process of automatic curation. Moreover, although biological ontologies and controlled vocabularies are meant to represent a wide range of concepts, they are not designed to reflect the exact wording found in the scientific literature. Therefore, some initial filtering of potential terms was necessary before they could be included in the BioLexicon. As an example, terms of proteins identified in the course of high-throughput experiments such as *hypothetical protein* were ignored due to their low information value. Also, a small number of highly ambiguous terms such as generic enzyme names were manually annotated as such. Other indications of a term's discriminatory power available in the BioLexicon include its frequency in Medline and the British National Corpus, as they have proven useful in the task of named entity recognition (Pezik *et al*., 2008).

The choice of these types of terms can be explained in two ways. Firstly, we felt it necessary to include the most common semantic types relevant to the biology domain, such as terms denoting gene and protein names, as well as terms for chemicals of biological interest or species

names. Secondly, including the smaller and more focused sets for terms such as operon names or sequence ontology terms was motivated by the intention to provide links from the BioLexicon to the Gene Regulation Ontology (Beisswanger *et al*., 2008) and make it suitable for text mining applications dealing with gene regulation topics.

4 General language words

To cover general language words that are used in biology, we have adopted words from the MedPost dictionary. This is distributed as a part of the MedPost POS tagger package and is available copyright free.[1] The dictionary consists of words appearing in MEDLINE abstracts.

The following numbers of entries were generated.

- 496 verbs (2,976 inflectional forms)
- 2,316 adjectives (2,385 inflectional forms)
- 428 adverbs (440 inflectional forms)
- 5,012 nouns (6,182 inflectional forms)

Inflections produced for verbs from the MedPost dictionary are the same as for terminological verbs. The POS types NN and NNS were assigned to the singular and plural forms of nouns, respectively.

Comparative and superlative forms of adjectives and adverbs were completed on the basis of the MedPost dictionary entries.

Since that dictionary was created for the purposes of a statistical POS tagger for the biomedical domain, it is incomplete from a linguistic point of view. For example, *common* and *commonest* are accommodated by the dictionary; however, *commoner* is not. Therefore, inflections of words in the dictionary were manually curated and added to the BioLexicon.

5 Biological term variants extracted from text

In addition to biomedical terms gathered from existing databases, the lexicon accommodates new variants of gene/protein names extracted from text.

[1] ftp://ftp.ncbi.nlm.nih.gov/pub/lsmith/MedPost/medpost.tar.gz

Table 1 NER performance

		R	P	F
Sequential labeling	Full	79.85	68.58	73.78
	Left	84.82	72.85	78.38
	Right	86.60	74.37	80.02

The extraction process consists of two steps. The first step identifies gene/protein names in text. Then, the second step maps new variants to existing entries.

This section provides a brief summary of the named entity recognition (NER) and term normalization used to populate the lexicon with gene/protein names extracted from biomedical literature.

5.1 Named Entity Recognition

For NER, we used our dictionary-based statistical named entity recognition tool (Sasaki *et al*., 2008).

The tool was trained with Conditional Random Fields (CRFs) (Lafferty *et al*., 2001) on the JNLPBA-2004 training data (Kim, 2004) and the Genia corpus (version 3.02) (Kim *et al*., 2003).

The test data used is the JNLPBA-2004 test set, which is a set of tokenized sentences extracted from 404 separately collected MEDLINE abstracts, where the term class labels were manually assigned, in accordance with the annotation specification of the Genia corpus.

Following the data format of the JNLPBA-2004 training set, our training and test data use the IOB2 labels, which are "B-protein" for the first token of the target sequence, "I-protein" for each remaining token in the target sequence, and "O" for other tokens. The window size was set to ±2 tokens of the current token.

Table 1 shows the evaluation results. Results are expressed according to recall (R), precision (P), and F-measure (F), which here measure how accurately the various experiments determine the left boundary (Left), the right boundary (Right), and both boundaries (Full) of protein names. The The F-score of the model trained with all the features was 73.78, which is the second best score for protein name recognition among research reported using the standard JNLPBA-2004 data set.

Gene/protein names identified by CRF classifiers with a probability greater than 99% are

selected as new gene/protein variant candidates from 15 million MEDLINE abstracts.

5.2 Term mapping

Terms automatically extracted from text were mapped to existing gene/protein name entries, which are given standard semantic identifiers called UniProt Accession Numbers. For efficiency reasons, term mapping was conducted through term normalization. Since the lexicon contains about two million gene/protein names, straightforward similarity calculation of term pairs is not practical: when an NER component extracts tens of millions of gene/protein name candidates from a corpus, the similarity distance of $2 \cdot 10^{13}$ pairs of terms must be calculated. This amount of computation can be drastically reduced to 10^7 normalizations and index lookups.

The normalization steps are as follows:

1. Create an inverse index that maps normalized forms to UniProt Accession Numbers.
2. Normalize newly extracted terms.
3. Lookup the inverse index to find UniProt Accession Numbers of the new terms.

There are several ways to normalize biomedical terms. We employed a method (Tsuruoka *et al*., 2007) where the normalization rules were automatically generated from a dictionary in which terms are clustered according to UniProt Accession Numbers. A brief summary of the method is as follows:

The method finds string-rewriting rules one by one based on the following complexity measure:

$$(\text{complexity}) = (\text{ambiguity}) \times (\text{variability})^{\alpha}$$

where the ambiguity quantifies how ambiguous the terms are in the dictionary, the variability value quantifies how variable the terms are, and α is the constant that determines the trade-off between ambiguity and variability.

Finding string rewriting rules is quite straightforward. We can represent any pair of terms *x* and *y* as follows:

$x = LXR$
y = LYR

where *L* is the left common substring shared by strings *x* and *y*, *R* is the right common substring, and *X* and *Y* are the substrings in the center that are not shared by the two strings. From this representation, we create the rule that replaces *Y* with *X*, which will transform *y* into *x*.

According to the experimental results reported in Tsuruoka *et al*. (2007), normalization performance is the same as normalization rules hand-crafted by domain experts. We generated 1,000 normalization rules, using the gene/protein names gathered from existing databases as the dictionary for normalization rule generation.

Terms mapped to more than 10 accession numbers are considered too ambiguous and filtered out from the new variant list. As a result, 70,105 variants of gene/protein names were extracted from 15 million MEDLINE abstracts.

6 Biomedical usages

In the lexicon, terminological verbs are linked to verb subcategorization frames (SCFs) which were acquired through unsupervised automatic acquisition techniques from linguistically pre-processed domain corpora. In the biomedical field, there is a strongly-felt desideratum that subcategorisation patterns should include strongly selected modifiers (such as location, manner and timing), as these are deemed to be essential for the correct interpretation of texts (Tsai *et al*., 2007). According to this, we adopted a "discovery" approach to SCF acquisition based on a looser notion of SCFs, which include typical verb modifiers in addition to strongly selected arguments.

In order to meet this basic requirement, a deep level of syntactic annotation was selected as the starting point for SCF induction. For this purpose, we used the Enju syntactic parser for English (Miyao *et al*., 2003)[1], characterised by a wide-coverage probabilistic HPSG grammar and an efficient parsing algorithm, and whose output is returned in terms of predicate-argument relations. In particular, we used the Enju version adapted to biomedical texts (Hara *et al*., 2005).

The SCF induction process was performed through the following steps:

[1] http://www-tsujii.is.s.u-tokyo.ac.jp/enju/

1. syntactic annotation of the acquisition corpus with Enju (v2.2). The acquisition corpus included both MEDLINE abstracts and full papers containing a total of approximately 6 million word tokens;
2. for each verbal occurrence, extraction of the observed dependency sets (ODSs). Note that the order of the dependencies in each ODS is normalised and does not reflect their order of occurrence in context;
3. induction of relevant SCF information associated with a given verb.

For each observed dependency set, the conditional probability given the verb type v was computed: thresholding was used, to filter out noisy frames (i.e., frames containing not only arguments and strongly selected modifiers, but also adjuncts) as well as possible errors of either parsing or ODS extraction. An ODS with an associated probability score beyond a certain threshold is selected as eligible SCF for that verb type.

Careful analysis of acquired SCFs revealed that many of the strongly selected modifiers were spread over different frames and that, even by lowering filtering thresholds, they either disappeared from the final output or their role was radically underestimated. We thus decided to complement acquired SCF information with information about individual dependencies of verbs. To detect typical verbal dependencies, corresponding to either arguments or strongly selected modifiers, we used the log likelihood score (henceforth ll (Dunning, 1993)). This is a logarithmic measure of the degree of correlation between v and each dependency type, gauged by comparing their joint probability with the probability of finding them together by chance, given their independent marginal distributions.

Due to the observed complementarity between acquired SCF and individual dependency information and its potential usage in different text mining applications, we decided to include both information types in the lexicon. SCF and dependency information was acquired for 759 orthographic variants of different terminological verbs, corresponding to 658 different base forms (see section 3.1). In particular, the lexicon includes 1,410 verb-SCF associations, involving 97 different SCF types, and 1,718 verb-dependency associations, involving 44 dependency types. For each SCF, the following information types are specified: its conditional probability given the verb, and the percentage of times it occurs with the verb in the passive voice. This latter information type is particularly useful to account for SCFs typically associated with the verb used in the passive voice: this is the case, for instance, of the verb *find* whose frame ARG1#ARG2#TO-INF# is typically (*i.e.*, 89% of the time) associated with passive contexts (*e.g., This was found to be interesting*). Concerning individual dependencies, the lexicon includes information about its association with respect to the verb, expressed in terms of the ll score, and – again – the percentage of times it occurs with the verb in the passive voice. Tables 2 and 3 show examples of subcategorization information stored in the lexicon for the verb *acquire.*

Table 2 Subcategorization frame examples

v	SCF	p(SCF|v)	% pass
acquire	ARG1#ARG2#	0.5461	0.1284
acquire	ARG1#ARG2#PP-in#	0.0886	0.0833
acquire	ARG1#ARG2#PP-from#	0.0406	0.1818
acquire	ARG1#ARG2#PP-by#	0.0406	0.0000
acquire	ARG1#ARG2#PP-during#	0.0295	0.3750

Table 3 Subcategorization slot examples

v	DEP	ll	% pass
acquire	ARG2#	579.96392	0.1512915
acquire	WH-when#	25.703417	0.1
acquire	PP-from#	22.716082	0.3333333
acquire	PP-by#	13.626654	0
acquire	PP-in#	13.416025	0.1666667

7. Comparison to existing lexicons

Several existing large-scale dictionaries and lexicons accommodate biological terms. Among them, many researchers use WordNet and the Specialist Lexicon for their text processing. WordNet is a general English resource which contains domain specific terms. The Specialist Lexicon was created by the National Library of Medicine, targeting the biomedical domain in general.

This section shows that our lexicon complements these popular lexical resources, by focusing on the words and relations that are

covered by our lexicon but not by these existing ones.

7.1 WordNet

WordNet (Fellbaum, 1998) is a general English thesaurus which additionally covers biological terms. We used WordNet 3.0[1] to evaluate term coverage.

Figure 2 shows the proportion of terminological words and relations (such as the word *retroregulate* and the relation *retroregulate* → *retroregulation*) in our lexicon that are also found in WordNet.

Since WordNet is not targeted at the biology domain, many biological terms and derivational relations are not listed.

7.2 UMLS Specialist Lexicon

The Specialist Lexicon[2] is a syntactic lexicon of biomedical and general English words, providing linguistic information about individual vocabulary items (Browne *et al*., 2003). Whilst it contains a large number of biomedical terms, our lexicon is tailored to the biology domain and covers more terms used within the biology domain, especially the molecular biology domain, than the Specialist Lexicon.

Figure 3 shows the proportion of words in our lexicon that are covered by the Specialist Lexicon.

Because the Specialist Lexicon is a biomedical lexicon and the target is broader than our lexicon, some biology-oriented words and relations are missing. For example the Specialist Lexicon includes the term *retro-regulator* but not *retro-regulate.* This means that derivational relations of *retro-regulate* are not covered by the Specialist Lexicon.

8. Conclusion and remarks

This paper has presented the BioLexicon, a unique resource comprising rich linguistic information suitable for bio-text mining applications. The lexicon has the following types of entries.

(1) Terminologies
(2) Derivational relations
(3) General English words
(4) Verb subcategorization frames

Comparisons with WordNet and the NLM Specialist Lexicon reveal that the BioLexicon covers words and relations which are pertinent to the biology domain but not included in these resources. We believe that it is a unique resource within the domain, which will play a complementary role to existing lexicons and thesauri.

The BioLexicon is available for non-commercial purposes under the Creative Commons license.

Our future work includes incorporating semantic event frames, such as gene regulation event frames, in the lexicon. Extrinsic evaluations of the lexicon in information extraction and question answering tasks are also planed.

Acknowledgement

This research is supported by EC IST project FP6-028099 (BOOTStrep), whose Manchester team is hosted by the JISC/BBSRC/EPSRC sponsored National Centre for Text Mining. The authors would like to thank Philip Cotter and Yoshimasa Tsuruoka for their assistance with the production of the lexical items. The authors also would like to thank Alessandro Lenci, Simone Marchi and Vito Pirrelli who contributed to the subcategorization extraction task.

References

Ananiadou, Sophia and John McNaught, editors. 2006. *Text Mining for Biology and Biomedicine*. Artech House, Norwood, MA.

Beisswanger, E., V. Lee, JJ Kim, D. Rebholz-Schuhmann, A. Splendiani, O. Dameron, S. Schulz, and U. Hahn. 2008. Gene Regulation Ontology (GRO): Design Principles and Use Cases. *Studies in health technology and informatics.* 136:9-14.

Browne, A.C., G. Divita, A.R. Aronson, and A.T. McCray. 2003. UMLS Language and Vocabulary Tools. In *Proc. of AMIA Annual Symposium 2003*, p.798.

Comrie, Bernard and Sandra A. Thompson. 2007. Lexical Normalization. In Timothy Shopen, editor, *Language Typology and Syntactic Description: Grammatical Categories and the Lexicon.* Chapter 8. Cambridge University Press.

[1] http://wordnet.princeton.edu/3.0/WordNet-3.0.tar.gz
[2] http://SPECIALIST.nlm.hih.gov

Dunning, T. 1993. Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics*, 19(1):61-74.

Fellbaum, C., editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA..

Francopoulo, G., M. George, N. Calzolari, M. Monachini, N. Bel, M. Pet, and C. Soria. 2006. Lexical Markup Framework (LMF). In *Proc. of LREC 2006*, Genova, Italy.

Hara, Tadayoshi, Yusuke Miyao, and Jun'ichi Tsujii. 2005. Adapting a Probabilistic Disambiguation Model of an HPSG Parser to a New Domain. In *Proc. of IJCNLP*, pages 199-210.

Kim, J-D., T. Ohta, Y. Tateisi, and J. Tsujii. 2003. GENIA Corpus - Semantically Annotated Corpus for Bio-Text Mining. *Bioinformatics*, 19:i180-i182.

Kim, J-D., T. Ohta, Y. Tsuruoka, Y. Tateisi, and N. Collier. 2004. Introduction to the Bio-Entity Recognition Task at JNLPBA, In *Proc. of the Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, pages 70-75.

Lafferty, J., A. McCallum, and F. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labelling Sequence Data. In *Proc. of the Eighteenth International Conference on Machine Learning (ICML-2001),* pages 282-289.

Miyao, Yusuke and Jun'ichi Tsujii. 2003. Probabilistic modeling of argument structures including non-local dependencies. In *Proc. of the Conference on Recent Advances in Natural Language Processing (RANLP 2003)*, pages 285-291.

Pezik, P., A. Jimeno, V. Lee, and D. Rebholz-Schuhmann. 2008. Static Dictionary Features for Term Polysemy Identification. In *Proc of LREC-08 Workshop on Building and Evaluating Resources for Biomedical Text Mining*.

Quochi, Valeria, Monica Monachini, Riccardo Del Gratta, and Nicoletta Calzolari. 2008. A Lexicon for Biology and Bioinformatics: the BOOTStrep Experience. In *Proc. of Language Resources and Evaluation Conference (LREC-08)*, pages 28-30.

Santorini, Biatrice. 1990. *Part-of-Speech Tagging Guidelines for Penn Treebank Project.* 3rd Revision, 2nd Printing. ftp://ftp.cis.upenn.edu/pub/treebank/doc/tagguide.ps.gz)

Sasaki, Yutaka, Yoshimasa Tsuruoka, John McNaught, and Sophia Ananiadou. 2008. How to Make the Most of NE Dictionaries in Statistical NER. *ACL-2008 Workshop on Current Trends in Biomedical Natural Language Processing (BioNLP-08)*, pages 63-70.

Smith, L., T. Rindflesch, and W. J. Wilbur. 2004. MedPost: a Part-of-Speech Tagger for BioMedical Text. *Bioinformatics,* 20:2320-2321.

Tjong Kim Sang, Erik F. and J. Veenstra. 1999., Representing Text Chunks. In *Proc. of the Ninth Conference of the European Chapter of the Association for Computational Linguistics (EACL-99),* pages 173-179.

Tsai, R.T-H., W-C. Chou, Y-S. Su, Y-C. Lin, C-L. Sung, H-J. Dai, I. T-H. Yeh, W. Ku, T-Y. Sung, and W-L. Hsu. 2007. BIOSMILE. *BMC Bioinformatics*, 8:325.

Tsuruoka, Yoshimasa, John McNaught, Jun'ichi Tsujii, and Sophia Ananiadou. 2007. Learning String Similarity Measures for Gene/Protein Name Dictionary Look-up Using Logistic Regression. *Bioinformatics*, 23(20):2768-2774.

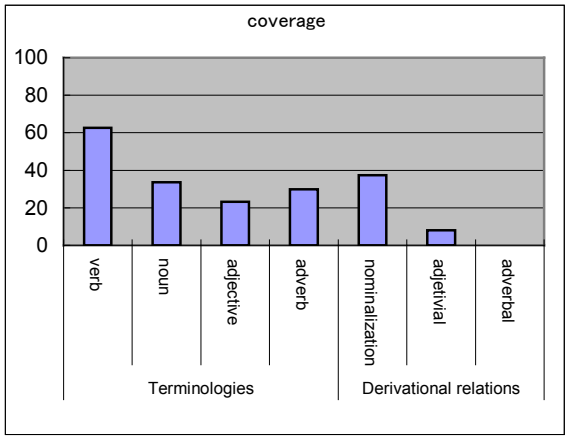


Figure 2 Word and relation coverage (%) in WordNet

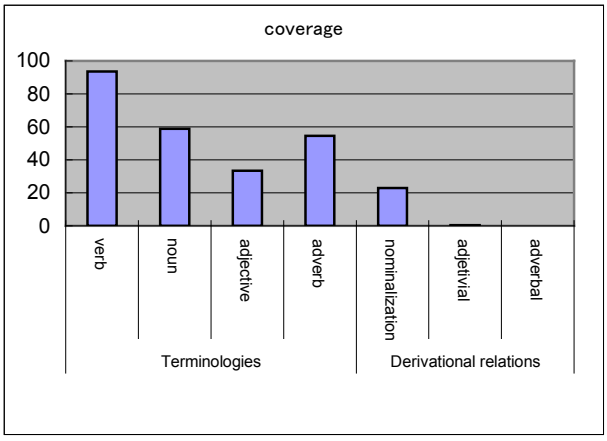


Figure 3 Word and Relation Coverage (%) in the Specialist Lexicon

Exploring the Compatibility of Heterogeneous Protein Annotations Toward Corpus Integration

Yue Wang* **Jin-Dong Kim*** **Rune Sætre*** **Jun'ichi Tsujii*†‡**
*Department of Computer Science, University of Tokyo
†School of Informatics, University of Manchester
‡National Center for Text Mining
Hongo 7-3-1, Bunkyo-ku, Tokyo 113-0033 JAPAN
{wangyue, jdkim, rune.saetre, tsujii}@is.s.u-tokyo.ac.jp

Abstract

We explore the sources of incompatibility between the protein annotations made to two corpora: GENIA and AIMed. We first hypothesize a problem with the incompatibility caused by corpus integration, and we measure the effect of the incompatibility on protein mention recognition. Through a series of experiments, we find several sources of the incompatibility, and suggest that more than half of the incompatibilities can be reduced by properly considering the scope of the annotated proteins, text preprocessing, and boundary annotation conventions.

1 Introduction

Human-annotated corpora are widely used in developing language processing systems. For biotext mining, there are several well-known corpora with protein mention annotations: GENIA (Kim et al., 2003), PennBioIE (Mandel, 2006), GENETAG (Tanabe et al., 2005), AIMed (Bunescu et al., 2005), etc. Based on these corpora, many protein mention recognizers have been developed, some of which report state-of-the-art performance (Wilbur et al., 2007).

However, there remains a well-known, but less studied, problem. Since the protein annotations are made by different groups, it is likely that the annotations in different corpora are not compatible with each other.

The incompatibility brings about several significant problems. For example, it is difficult to effectively utilize more than one corpus to develop a protein mention recognizer. Indeed, there has never been a protein recognizer developed by utilizing multi-corpora, because it is hardly possible to benefit from corpus integration. It is also difficult to compare systems developed with different corpora. Although there are many systems that claim to recognize protein mentions from MEDLINE texts, their reported performance varies significantly (Tsai et al., 2006). The mentioned problems are largely caused by the incompatibility of different protein annotations, and can not be solved effectively without understanding the differences in the annotations (Pyysalo et al., 2008).

In this paper, we explore the potential sources of incompatibility between two well-known corpora with protein annotations: GENIA and AIMed. We first characterize the incompatibility resulting from using the two corpora as a single resource. Then, we carefully study the documentation of the two corpora in order to figure out the sources of incompatibility. Through a series of experiments, we explore the possible sources, while finding reasonable ways to avoid the problems caused by the incompatibility of protein annotations. Experimental results show that it is feasible to reduce the incompatibility of the heterogeneous annotations by properly considering the differences. Meanwhile, we can get a comprehensive understanding of the two corpora, and take advantage of the annotations in both corpora, while minimizing the negative effects caused by their inconsistency.

The paper is organized as follows. In section 2, the two corpora used for exploration, GENIA and AIMed, are described. Two preliminary experiments characterizing the problem of combining two incompatible corpora are reported in section 3. From section 4 to section 6, the corpora's differences are explored regarding three aspects: the scope of the entities of interest, text preprocessing, and the conventions for boundary decisions,

respectively. We propose a way to reduce the corpus inconsistency for each aspect. Following a final experiment on the remaining inconsistencies in section 7, our research is concluded in section 8.

2 Data

Here, we briefly introduce the GENIA and the AIMed corpora, focusing on their size and covered domain.

2.1 The GENIA corpus

The GENIA corpus (version 3.02) is a collection of articles extracted from the MEDLINE database with the MeSH terms "human", "blood cells" and "transcription factors". There are 2,000 abstracts and 18,545 sentences totally. The term annotation is according to a taxonomy of 48 classes based on a chemical classification. Among the classes, 36 terminal classes were used to annotate the corpus. The total number of annotated terms is 93,293.

In recent years, the GENIA corpus has become one of the most frequently used corpora in biomedical named entity recognition (Bio-NER) task (Cohen and Hersh, 2005).

2.2 The AIMed corpus

The AIMed corpus consists of 225 MEDLINE abstracts, of which 180 are known to describe interactions between human proteins, while the other 45 do not refer to any interaction. In all, there are 1,969 sentences and 4,084 protein references.

The AIMed corpus is now one of the most widely used corpora with protein interaction annotation. Its protein annotations are parts of the protein interaction annotations.

3 Preliminary experiments

We performed two preliminary experiments in order to confirm the following two assumptions. First, we can improve the performance of a protein mention recognizer by increasing the size of the training data set. Second, the system performance will drop when incompatible annotations are introduced into the training data set. The protein mention recognizer used in our work is a Maximum Entropy Markov Model n-best tagger (Yoshida and Tsujii, 2007). To reduce our task to a simple linear sequential analysis problem, we removed all the embedded tags in GENIA and AIMed, and only retained the outermost tags.[1]

We divided the AIMed corpus into two parts, 70% for training and the remainder for testing. In the first experiment, we only used the AIMed training part. In this experiment, we performed seven sub-experiments, and each time, we added 10% more abstracts into the training portion. In the second experiment, besides the AIMed training part, we also added the GENIA protein annotations. In both experiments, we performed the evaluations on the AIMed test part according to the exact matching criterion. In this paper, all the evaluations are carried on the AIMed test part, whose size is 30% of the AIMed corpus. For convenience, the AIMed training part is simply called the "AIMed corpus" in the following.

A learning curve drawn from the results of the two mentioned experiments is shown in Figure 1.

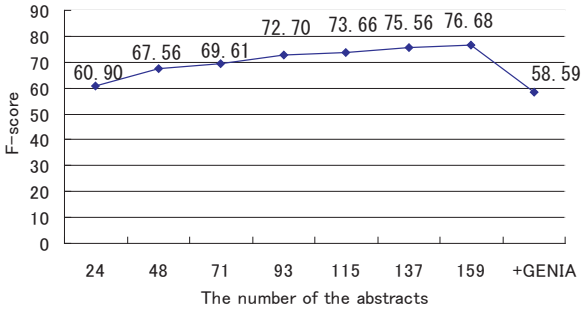


Figure 1: Learning curve drawn from the results of two preliminary experiments.

We can see that the learning curve is still increasing when we used up all the training portions of the AIMed corpus. We would expect a further improvement if we could add more training data in a large scale, e.g. the GENIA corpus, which is ten times bigger than the AIMed corpus. But when we actually add the protein annotations in the GENIA corpus to the training data set, we witness a drastic degradation in the performance. We assume that the degradation is caused by the incompatibility of the protein annotations in the two corpora, and we further assume that as the incompatibility decreases, the learning curve would get back to the original increasing direction.

In the following three sections, we will ex-

[1] There are 136 embedded occurrences in the AIMed corpus, 3 of which are triple-nested. And there are 1,494 embedded cases in the GENIA corpus.

plain some differences that cause the performance degradation, both from the perspective of documentation and from the experimental results. According to the differences, we design a series of experiments to reduce the incompatibility between the two corpora when using them in an integrated way.

4 Scope of the entities of interest

Although both corpora include protein mention annotations, the target tasks are different. The GENIA annotation centers on mining literature for general knowledge in biology, while the AIMed annotation focuses on extracting interactions among individual proteins. The difference has affected the scope of annotated proteins: *GENIA concerns all the protein-mentioning terms, while AIMed focuses only on references of individual proteins.*

4.1 Categories of annotated proteins

The scope of the proteins annotated in the GENIA corpus is defined in the GENIA ontology (Ohta et al., 2002); besides the protein class, other classes such as *DNA*, *RNA*, *cell_line* and *cell_type* are also included. Further, the protein class is categorized into seven subclasses: *Protein_complex*, *Protein_domain_or_region*, *Protein_family_or_group*, *Protein_molecule*, *Protein_substructure*, *Protein_subunit* and *Protein_ETC*. In other words, in GENIA, the protein is defined to include all seven concepts. No other protein subclasses are defined in the GENIA corpus.

In the case of AIMed, the scope of the proteins annotated is described by the following statement in the tagging conventions: generic protein families are not tagged, only specific names (protein molecules) that could ultimately be traced back to specific genes in the human genome are tagged. E.g. "Tumor necrosis factor" would not be tagged, while "tumor necrosis factor alpha" would be.

Hence, the documentation of the two corpora explicitly states that:

(1) the mentions of protein families (*Protein_family_or_group*) are annotated in GENIA, but not in AIMed, and

(2) individual proteins (*Protein_molecule*) are annotated in both corpora.

4.2 Compatible annotations

Section 4.1 provided two clues for the inclusion/exclusion of *Protein_molecule* and *Protein_family_or_group* annotations, specified in the published literature. However, there are five other protein subcategories annotated in GENIA, and we could not find any mentions regarding the inclusion or exclusion of the five protein subcategories in the scope of the annotations in AIMed. We performed a series of experiments to confirm the two clues that we found, and to find other clues for the other five protein subclasses.

+ Subcategory	Recall	Precision	F-score
molecule	52.87	82.80	64.54
subunit	29.63	86.57	44.15
ETC	28.61	89.60	43.37
substructure	28.10	88.00	42.59
complex	28.48	79.93	42.00
domain_or_region	27.71	79.49	41.10
family_or_group	26.82	65.02	37.97

Table 1: Experimental results of the AIMed corpus plus the GENIA protein subcategory annotations.

We used each of the GENIA protein subclasses in turn together with the AIMed corpus for the training. That is, each time we regarded the annotations from a different GENIA protein subclass as positive examples. The experimental results are listed in Table 1, showing the exact matching scores. According to the table, it is most harmful to add the *Protein_family_or_group* annotations, supporting the clue we have already found: the mentions of protein families are annotated in GENIA, but not in AIMed. Also, we notice that the GENIA *Protein_molecule* annotations least negatively affect the performance of recognizing the proteins tagged in the AIMed corpus, and the *Protein_subunit* and *Protein_complex* follow it[2]. Meanwhile, we observe that by adding the protein subcategory annotations, the precision of the protein mention recognition on the AIMed corpus is very good, while the recall is very low. This observation suggests that if we add the annotations of the three protein sub-classes into the training material at the same time, we could improve the recall while maintaining good precision. Table 2 shows the experimental results based on this

[2]Because the number of the *Protein_substructure* annotations and the *Protein_ETC* annotations are very small (103 and 85, respectively), the two protein subcategories were excluded from consideration.

AIMed + Subcategory	Criterion	Recall	Precision	F-score
molecule + subunit	Exact	53.77	80.96	64.62
	Left	58.75	88.46	70.61
	Right	56.70	85.38	68.15
	Overlap	62.20	93.65	74.75
molecule + subunit + complex	Exact	54.15	76.40	63.38
	Left	62.58	88.29	73.24
	Right	57.34	80.90	67.12
	Overlap	67.05	94.59	78.47

Table 2: Experimental results of the AIMed corpus plus the annotations of three GENIA protein subcategories.

Training data	Criterion	Recall	Precision	F-score
AIMed	Exact	74.33	79.18	76.68
	Left	78.93	84.08	81.42
	Right	76.63	81.63	79.05
	Overlap	81.48	86.80	84.06
AIMed + GENIA_Protein	Exact	56.19	61.20	58.59
	Left	66.79	72.74	69.64
	Right	59.90	65.23	62.45
	Overlap	72.80	79.28	75.54

Table 3: Experimental results of the AIMed corpus and the AIMed corpus plus the GENIA protein annotations.

hypothesis. In addition to the exact, left boundary and right boundary matching criteria, we also tested an overlap matching criterion (Franzén et al., 2002), namely, if any part of a protein mention is identified, it will be considered as a correct answer. The experimental results show that when we collectively use the GENIA annotations of the three protein subclasses, the recall improved significantly while minimizing decrease in precision. For fair comparison, we also applied the left boundary, right boundary and overlap matching criteria to the results gained by using the AIMed corpus, and the AIMed corpus plus the GENIA protein annotations, respectively. The results are shown in Table 3.

Since our goal is to find a way to make the learning curve go back to an increasing state, we set the performance induced from the pure AIMed corpus as the minimum goal. Then, the potential (maximum) reduction rate of incompatibility can be calculated by Formula (1):

$$R_e = \frac{F_e - F_{A+G}}{F_A - F_{A+G}}\%, \qquad (1)$$

where R_e denotes the corpus incompatibility reduction rate of a given experiment, F_e denotes the F-score of the given experiment, F_A and F_{A+G} denote the F-score of the training with the AIMed corpus, and with the AIMed corpus plus the GENIA protein annotations, respectively.

We can say that, by combining the GENIA *Protein_molecule*, *Protein_subunit* and *Protein_complex* annotations with the AIMed corpus, we reduced the corpus incompatibility by 30.56% (the left boundary matching criterion[3]). So, when we want to introduce the annotations from the GENIA corpus, we can use the annotations of the three protein subclasses. It further indicates that the annotations of these three protein subclasses in both corpora are compatible to some extent.

We found sentences including *Protein_subunit* or *Protein_complex* annotations, which will not cause the incompatibility during corpus combination[4]. That is, in both corpora, these entities are regarded as proteins, so we can introduce most of the GENIA annotations of these entities into AIMed, without negative influence. Some examples are shown in Figure 2. For comparison, all the entity annotations are shown in the figure.

4.3 Ambiguity between DNAs and genes

The protein annotations in the AIMed corpus include not only proteins, but also genes, without differentiating them. In the case of the GENIA corpus, the protein annotation is applied

[3]To avoid underestimation, we adopted a looser criterion.

[4]*Protein_molecule* has already been annotated in both corpora.

Disruption of the **Jak1 binding proline-rich Box1 region** (*Protein_domain_or_region*) of **IL-4Ralpha** (*Protein_subunit*) abolished signaling by this **chimeric receptor** (*Protein_family_or_group*). (GENIA PMID 9159166)

Only weak **IL-13** binding activity was found in cells transfected with only **IL-13Ralpha**; however, the combination of both **IL-13Ralpha** and **IL-4Ralpha** resulted in substantial binding activity, with a Kd of approximately 400 pM, indicating that both chains are essential components of the **IL-13 receptor**. (AIMed PMID 8910586)

Triflusal and HTB may exert beneficial effects in processes in which de novo **COX-2** (*Protein_molecule*) expression is involved and, in a broader sense, in pathological situations in which genes under **nuclear factor-kappaB** (*Protein_complex*) control are up-regulated. (GENIA PMID 10101034)

In this review, we summarize these and other TNF receptor-associated proteins and their potential roles in regulating the activation of **nuclear factor-kappaB** and apoptosis, two major responses activated by engagement of TNF receptors by the ligand. (AIMed PMID 9129204)

Figure 2: Sentences including the same annotated entities. (The boldface represents an annotated entity and in the GENIA examples the word under the line represents the class used to annotate the entity.)

only to proteins, while genes are annotated in the scope of DNA annotations. This suggests that it would improve the consistency if we treat gene annotations in the GENIA corpus in the same way as done in the AIMed corpus. However, the GENIA annotation does not include an explicit gene annotation. Instead, genes are annotated as instances of *DNA_domain_or_region* which is also applied to other DNA regions; e.g. binding sites and c-terminals. We assume that if the *DNA_domain_or_region* annotations that are not pure genes can be filtered out from all the *DNA_domain_or_region* annotations, we can find some examples from the remaining GENIA *DNA_domain_or_region* annotations that will positively affect the corpus combination. Therefore, if we assume that the performance of the recognizer trained with the AIMed corpus is good enough,[5] it will find most of the gene mentions in the GENIA corpus. The true positives, which are annotated as *DNA_domain_or_region* in the GENIA corpus and are also recognized by the recognizer, will include *DNA_domain_or_region* instances which are genes.

To examine the performance of the filtering, we added all the *DNA_domain_or_region* annotations to the training set in one experiment, and only the "true positive" classified "genes" in another experiment. The results shown in Table 4 indicate that the disambiguation between DNAs and genes works, although the improvement degree resulting from the filtering is not big.

As mentioned in section 4.2, adding only the *Protein_molecule*, *Protein_subunit* and *Protein_complex* annotations gives the best performance on the AIMed test part. Next, besides these three annotation types, we also added the filtered *DNA_domain_or_region* annotations to train our protein mention recognizer. The experimental results are shown in Table 5. Compared with Table 3, the corpus incompatibility is reduced 40.58% by adding the filtered D*NA_domain_or_region* annotations (the left boundary matching criterion).

AIMed + Subcategory	Recall	Precision	F-score
DNA	29.76	80.62	43.47
DNA_which_is_a_gene	30.27	84.95	44.63

Table 4: Experimental results of the disambiguation between *DNA_domain_or_region* and gene based on the exact matching criterion.

Criterion	Recall	Precision	F-score
Exact	56.58	74.70	64.39
Left	65.39	86.34	74.42
Right	60.28	79.60	68.60
Overlap	70.63	93.25	80.38

Table 5: Experimental results of adding the *Protein_molecule*, *Protein_subunit* and *Protein_complex*, and the filtered *DNA_domain_or_region* annotations.

[5] Of course, the filtering would only work perfectly, on the premise that the performance of the recognizer is perfect, so it will be a rough filtering.

5 Text preprocessing

In the AIMed corpus, a pre-tokenization policy is taken, which is the Penn Tree Bank style tokenization. Hence, we also pre-tokenized the GENIA corpus according to the Penn Tree Bank style, and retrained our recognizer by combining the *Protein_molecule_subunit_complex* annotations, and the filtered *DNA_domain_or_region* annotations, with the AIMed corpus. The experimental results are shown in Table 6. Compared with the results from Table 3, we reduce the incompatibility of the two corpora by 44.57% (the left boundary matching criterion).

Criterion	Recall	Precision	F-score
Exact	58.75	75.29	66.00
Left	66.67	85.43	74.89
Right	61.56	78.89	69.15
Overlap	70.88	90.83	79.62

Table 6: Experimental results of taking Penn Tree Bank style pre-tokenization.

6 Boundary of protein mentions

Even though the scope of the proteins to be annotated is standardized, the boundary of the protein mentions is still ambiguous. In general, the boundary ambiguity often arises in two ways. One ambiguity exists in making general guidelines for which part of a text expression is in charge of mentioning a protein. The other ambiguity exists regarding the confusion concerning the application of these guidelines. The confusion can be measured by entropy, as described below.

6.1 Determining which part is in charge of protein mentions

For example, when the text expression "p21ras protein" is given, it is not obvious whether to annotate the word "protein" as a part of the protein mentioning expression or not. We found that GENIA includes the word "protein" in the protein mentioning expressions, while AIMed excludes it. If a Bio-NER system is trained with AIMed, and we evaluate this system on GENIA, we can see a boundary matching error in "p21ras" , where "protein" is not included in the tag. However, for a text mining system, this error may be acceptable, since the system has correctly identified "p21ras" as a protein, and this information is adequate to mine the relationship between "p21ras" and another protein. Similarly, "the p21ras protein" or "the p21ras" could also be considered correct.

This also affects the average length of the protein mentions in the two corpora. The average length per protein mention is 1.9 tokens in the AIMed corpus, and 2.9 in the GENIA corpus. The percentages of protein mentions over 3 tokens in AIMed and GENIA are 12.65% and 50.29%, respectively. Many long protein mentions are introduced when we add the GENIA annotations into AIMed; this is another source of the performance degradation of recognizing shorter protein mentions in the AIMed corpus.

6.2 Annotation entropy for boundary words

In a given corpus, some words are annotated as inside of protein mentions, while other words are not. The annotation entropy of boundary words is calculated by Formula (2). For the sake of brevity, the (boundary) "word" discussed in this subsection describes the word that appears at the beginning or end of an annotated entity, or that abuts an annotated entity. When the annotation entropy of a boundary word is 0, this word is perfectly annotated and keeps the annotation consistency in the entire corpus. On the contrary, when the annotation entropy of a boundary word is 1, this word is so disorderly annotated that we can hardly find any rules about whether to regard it as a part of protein mentions or not. The value of E_b ranges from 0 (consistent) to 1 (inconsistent).

$$E_b = -(P_a \log_2 P_a + \overline{P_a} \log_2 \overline{P_a}), \qquad (2)$$

where, E_b denotes the annotation entropy of a given word, P_a denotes the percent of the annotated occurrences of this word, and $\overline{P_a}$ denotes the percent of the occurrences of this word that are not annotated.

In general, there are two types of boundary words: descriptive adjectives (such as "normal" or "activated"), and nouns, denoting the semantic category, occurring either before (as modifiers, such as "human") or after (as heads, such as "protein" or "molecule") The GENIA tagger[6] was used to determine the words Part-Of-Speech. Some boundary words appearing in each corpus are listed in Table 7. In order to characterize the differences between the two corpora in terms of

[6] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/tagger/

Category	Word	AIMed			GENIA		
		N_a	N_n	E_b	N_a	N_n	E_b
Adjective	recombinant	1	7	0.54	36	24	0.97
	soluble	1	9	0.48	14	15	1.00
	inducible	0	0	0.00	18	18	1.00
	putative	0	0	0.00	15	15	1.00
	constitutive	0	0	0.00	12	11	1.00
	low	0	0	0.00	14	11	0.99
	major	0	0	0.00	25	15	0.95
Noun_before	protein	12	29	0.73	164	18	0.47
Noun_after	protein	36	17	0.96	749	45	0.31
	site	0	0	0.00	21	12	0.95

Table 7: List of boundary words. Here, Noun_before indicates the noun occurring before an entity as a modifier, Noun_after indicates the noun occurring after an entity as a head. N_a is the number of the annotated occurrences, and N_n is the number of not annotated occurrences.

annotation entropy of boundary words, the words with annotation entropy close to 1 in one of the two corpora were included in Table 7.

From the table, we can see that the boundary annotation problem appears for various words. The distribution of these words is diverse, especially for the case of adjectives. Since as few extra characters as possible were tagged in the AIMed corpus, only the names of protein mentions are annotated, and most of the adjectives are not annotated. However, in the GENIA corpus, the adjectives before protein mentions are annotated only if they are required for the meaning of protein mentions (e.g. in the protein mention of "inducible cAMP early repressor", "inducible" is annotated, because it is needed for the comprehension of the meaning.).

In this situation, we need an alternative matching criterion other than the exact matching. To provide alternative evaluation perspectives, researchers have developed a variety of evaluation criteria that relax the matching to different degrees. Here, as previously shown, in addition to exact matching, left boundary, right boundary, and overlap matching are considered. Thus, if we assume that the expected minimal performance of the F-score of this work is near 84.06% (no corpus integration), it can be said that the possible maximum reduction of the incompatibility between the two corpora by the methods in this paper is 56.81% (the overlap matching criterion in Table 5).

7 Experiments performed on the non-overlapped data

From our current best results shown in section 6, there are still remaining incompatibilities responsible for more than half of the total incompatibilities. Since the abstracts in the two corpora are collected in different ways, it is supposed that the proteins mainly mentioned in the two corpora are heterogeneous, resulting in the incompatibility.

To quantify this assumption, we counted the number of identical names between the training and the testing part of AIMed, and between the AIMed training part and the GENIA protein annotations. In order to find the unique proteins, we normalized the protein mentions; for example, we changed uppercases to lowercases, and removed punctuation marks, spaces, and the appositions in parentheses. There are 766 unique entities in the AIMed training part, 250 in the AIMed test part and 7,759 in the GENIA corpus. Between the AIMed training part and the GENIA protein annotations, there are merely 270 unique entities that are overlapped. Further, between the training and the test parts of AIMed, the number of the overlapped unique entities is just 91. Due to the low overlapping coefficient, we divided the AIMed test part into two parts: one includes the annotations overlapped with the AIMed training part, and another includes the annotations that are not overlapped with the AIMed training part. We re-evaluated our recognizer on the latter part. The experimental results are shown in Table 8. From the table, even though the performance on the non-overlapped part did not improve by adding the three GENIA protein subcategories and the

filtered *DNA_domain_or_region* annotations, and by taking pre-tokenization, the result is very close to the result gained by using only the AIMed corpus for training. It implies that the heterogeneity of the proteins in the two corpora is another major source of the incompatibility, and suggests that we need find an appropriate way to properly consider the heterogeneity.

Training data	Recall	Precision	F-score
AIMed	71.46	40.54	51.73
AIMed+GENIA	63.31	43.21	51.36

Table 8: Experimental results of the non-overlapped part based on the overlap matching criterion. The last row shows the result of adding the three GENIA protein subcategories and the filtered *DNA_domain_or_region* annotations, and taking pre-tokenization.

8 Conclusions

Incompatibility of protein annotations in different corpora is a well known, but less studied, problem. In order to measure the effect of the incompatibility on protein mention recognition, we performed an experiment of corpus integration, which showed a significant degradation of performance due to the incompatibility.

Motivated by the result of the preliminary experiment, we investigated the source of incompatibility through a series of experiments. The results were encouraging. We found three main sources of incompatibility: the scope of the entities of interest, text preprocessing, and boundary of protein mentions, thus suggesting ways of reducing or avoiding the incompatibility. Meanwhile, we could improve our understanding of the difference of the two corpora, leading to a better understanding about the performance of protein recognizers based on them.

Some future works will follow from two perspectives. In order to achieve an actual improvement of protein recognition by integrating different corpora, we will further investigate the remaining source of incompatibility, finding a suitable model to integrate heterogeneous annotations. In order to better understand the difference of protein annotations, we will extend the comparison work to other corpora, e.g. GENETAG, toward a better consensus of protein annotations.

Acknowledgments

This work was partially supported by Grant-in-Aid for Specially Promoted Research (MEXT, Japan) and Genome Network Project (MEXT, Japan).

References

Razvan Bunescu, Ruifang Ge, Rohit J. Kate, Edward M. Marcotte, Raymond J. Mooney, Arun K. Ramani and Yuk Wah Wong. 2005. Comparative Experiments on Learning Information Extractors for Proteins and their Interactions. *Artificial Intelligence in Medicine*, 33:139–155.

Aaron M. Cohen and William R. Hersh. 2005. A Survey of Current Work in Biomedical Text Mining. *Briefings in Bioinformatics*, 6:57–71.

Kristofer Franzén, Gunnar Eriksson, Fredrik Olsson, Lars Asker, Per Lidén and Joakim Cöster. 2002. Protein Names and How to Find Them. *International Journal of Medical Informatics*, 67:49–61.

Jin-Dong Kim, Tomoko Ohta, Yuka Tateisi and Jun'ichi Tsujii. 2003. GENIA Corpus - a Semantically Annotated Corpus for Bio-textmining. *Bioinformatics*, 19(Suppl. 1):i180–i182.

Mark A. Mandel. 2006. Integrated Annotation of Biomedical Text: Creating the PennBioIE Corpus. *in Proceedings of the Workshop on Text Mining, Ontologies and Natural Language Processing in Biomedicine*, Manchester, UK.

Tomoko Ohta, Yuka Tateisi, Hideki Mima and Jun'ichi Tsujii. 2002. GENIA Corpus: an Annotated Research Abstract Corpus in Molecular Biology Domain. *in Proceedings of the Human Language Technology Conference*, San Diego, USA.

Sampo Pyysalo, Antti Airola, Juho Heimonen, Jari Björne, Filip Ginter and Tapio Salakoski. 2008. Comparative Analysis of Five Protein-protein Interaction Corpora. *BMC Bioinformatics*, 9(Suppl 3):S6–S16.

Lorraine Tanabe, Natalie Xie, Lynne H Thom, Wayne Matten and W John Wilbur. 2005. GENETAG: a Tagged Corpus for Gene/Protein Named Entity Recognition. *BMC Bioinformatics*, 6(S1):S3–S9.

Richard Tzong-Han Tsai, Shih-Hung Wu, Wen-Chi Chou, Yu-Chun Lin, Ding He, Jieh Hsiang, Ting-Yi Sung and Wen-Lian Hsu. 2006. Various Criteria in the Evaluation of Biomedical Named Entity Recognition. *BMC Bioinformatics*, 7:92–99.

John Wilbur, Larry Smith and Lorrie Tanabe. 2007. BioCreative 2. Gene Mention Task. *in Proceedings of the Second BioCreative Challenge Evaluation Workshop*, Madrid, Spain.

Kazuhiro Yoshida and Jun'ichi Tsujii. 2007. Reranking for Biomedical Named-Entity Recognition. *in Proceedings of the workshop of the 45th Annual Meeting of the Association for Computational Linguistics*, Prague, Czech Republic.

Short papers

How Complex are Complex Protein-protein Interactions?

Jari Björne,[1] **Sampo Pyysalo,**[2] **Filip Ginter**[1] and **Tapio Salakoski**[1,2]
[1]Department of IT, University of Turku
[2]Turku Centre for Computer Science (TUCS)
Joukahaisenkatu 3-5
20520 Turku, Finland
firstname.lastname@utu.fi

Abstract

The extraction of protein-protein interactions (PPI) from text requires a formal PPI representation. We use the BioInfer and GENIA corpora to study two such representations: a "binary" interaction model consisting of pairs of proteins and a "complex" model where interactions are defined as a network of proteins and their relations. As both of these formats can be seen as graphs, we contrast them with syntactic dependency graphs, a common tool for PPI extraction. We find that unlike binary interactions, complex interactions closely resemble dependency parses, especially those in the Stanford scheme. We therefore argue that despite appearances, complex interactions might be easier to extract. We also notice the similarity between the independently developed BioInfer and GENIA interaction representations and the Stanford dependency scheme. This suggests an emerging consensus on the representation for complex PPI, supporting the value of these tools and resources for PPI extraction.

1 Introduction

Protein-protein interaction (PPI) extraction is a central, widely studied task in biomedical natural language processing. The simplest model of PPI, used in most corpora and extraction studies, represents each interaction as a pair of protein names. Several systems have been introduced for extracting such binary interactions, but considerable challenges remain (Krallinger et al., 2007).

Recently, two corpora with more detailed interaction annotation have been introduced: the BioInfer (Pyysalo et al., 2007) and GENIA Event corpora (Kim et al., 2008) annotate complex structured relations (Figures 1 and 2). These "complex interactions" differ from binary interactions in that they can have more than two arguments, and allow interactions as arguments, thus enabling annotation of complex nested relations such as in "A causes B to bind C". Complex interactions can also be thought of in terms of semantic frames, with the edges of the complex interaction corresponding to the arguments of a verb frame (Cohen and Hunter, 2006).

For BioInfer, this annotation has also been translated into binary interactions (Heimonen et al., 2008), providing an opportunity to compare complex and binary interactions. In addition to PPI annotation, both BioInfer and GENIA include syntactic annotation that can be accessed in various dependency representations. Dependency has been argued to be well suited for applications such as information extraction, and dependency parsing is both well studied and frequently applied in the biomedical domain (de Marneffe et al., 2006; Clegg and Shepherd, 2007).

We are not aware of methods that would aim to extract the complex interactions annotated in the BioInfer and GENIA Event corpora. Neither has the relationship between simple and complex PPI annotation been studied in detail. Our aim here is to explore this relationship and thus take a first step towards complex PPI extraction.

2 Analysis and Discussion

2.1 Complex vs. Binary Interactions

We first observe that both the "binary" and "complex" representations can be viewed as forms of semantic networks (graphs). In the former case protein nodes are connected by edges expressing interactions, in the latter, both proteins and words

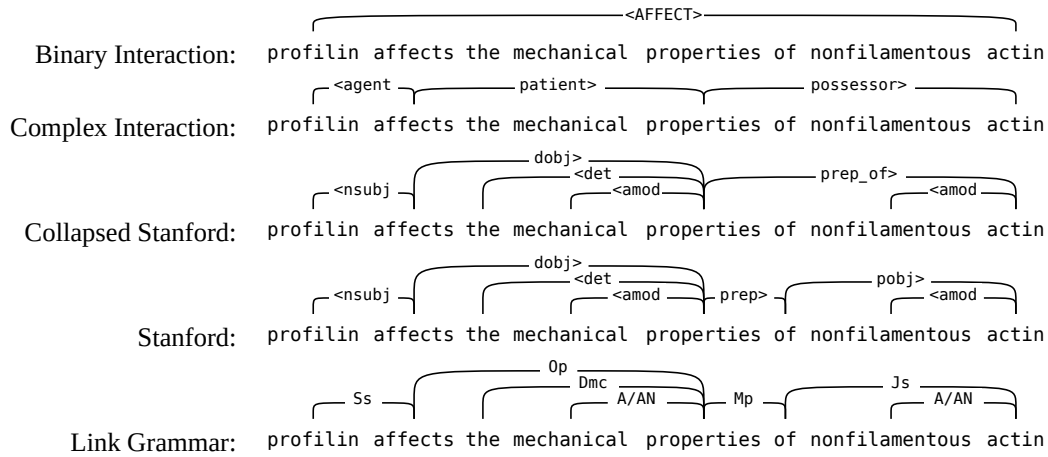


Figure 1: Example from the BioInfer corpus, with the interaction annotation and the three parse schemes

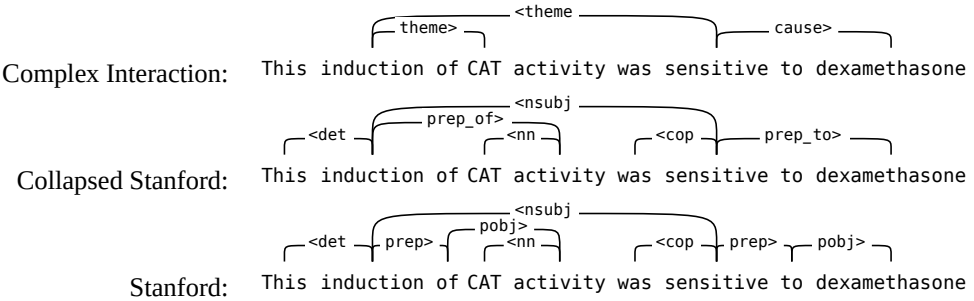


Figure 2: Example from the GENIA corpus, with an annotated interaction and two parse schemes.

stating their relations act as nodes and edges express their roles. Syntactic dependency parses are also graphs where words (nodes) are linked with dependencies (edges). Thus, as syntax and the binary and complex interactions have a graph representation, their relationships can be studied as a mapping between these graphs.

Figures 1 and 2 illustrate sentences from the BioInfer and GENIA corpora with their PPI annotation and dependency syntax, showing all the available graph representations (see Section 2.2 for descriptions of the syntactic annotations). Figure 1 shows a binary interaction between words which are several dependencies away from each other in the syntactic parses. On the other hand the graph of the complex interaction corresponds more closely to the dependency parse. This is typical: the words annotated as expressing interactions frequently fall on the shortest dependency path between the proteins. By providing intermediate nodes along the dependency path that connects the proteins involved in binary interactions, complex interactions subdivide the concept of "interaction" into smaller parts. As these simple relations can correspond better to syntactic features in a sentence, they could be easier to extract than the diverse binary interactions.

To study the feasibility of extracting complex interactions, we compared the dependency parse representations with both binary and complex interactions from the BioInfer corpus, and complex interactions from the GENIA corpus.

2.2 Processing the Corpora

The BioInfer and GENIA annotation schemes are designed to capture complex biological interactions in detail. The BioInfer format annotates e.g. entities and interactions with predicates appropriate for these tasks. The BioInfer annotation can be converted to several derived formats more suited for different uses. For these experiments, we transformed BioInfer into a semantic network representation in which the entire annotation of a sentence is defined as a directed graph (Heimonen et al., 2008). The edge labels define the semantic roles between entities and relations (e.g. *agent*, *patient*) and between different entities (e.g. *sub/super* for part/whole relations). Predicates not bound to text in the original annotation, such as most occurrences of *EQUAL* (an identity

relationship between entities), were converted to edges. To compare the complex interactions with the binary ones, we also used a binarised version of the annotation, where interactions are simple pairs of named entities. BioInfer has manually annotated dependency parses in the Link Grammar (Sleator and Temperley, 1991) and Stanford formats.

For GENIA, we used the recently published event annotation. This annotation has manually annotated complex interactions for 9372 sentences. This was interpreted as a semantic network with the edges labeled *theme* and *cause* as defined in the event annotation; no edges were derived from the entity annotation. From these, we selected the subset of 1968 sentences which had manually annotated parses in the beta version of GENIA Treebank (GTB) (Tateisi et al., 2005). The GTB annotation was converted into dependency which was collapsed with the software introduced by de Marneffe et al. (2006); we refer to this study for a description of the representation. We used the manually annotated gold standard parses for all evaluations.

2.3 Connecting interactions to parses

To compare semantic interaction annotation to dependency parses we have to map the interactions to the sentence text. This is done based on the *text bindings*, which connect the annotations to the words expressing them. However, in both BioInfer and GENIA these text bindings can consist of multiple words. For example when the entity *Acanthamoeba profilin* takes part in an interaction, the edge that links to it connects to this pair of words. By contrast, in a dependency parse, all edges connect to single words. Thus, for comparison with dependency parses, interaction edges connecting to multi-word entities are mapped to a single word. We used the Stanford parse to map these edges to syntactic head tokens.

2.4 Comparing interactions to parses

To see how closely complex interactions resemble a dependency parse, we measured the shortest path in the dependency graph between two tokens connected by an edge in the interaction graph. We compared the lengths of these shortest paths between the available three parses for BioInfer and the two for GENIA (Figure 3). We notice that complex interaction edges most likely

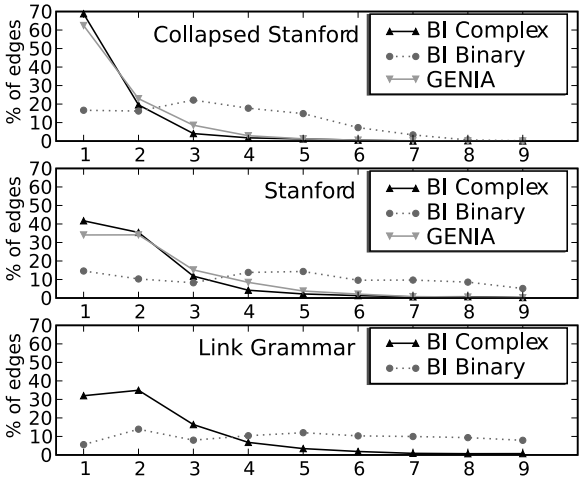


Figure 3: Percentage of interaction edges plotted against the length of the shortest path of dependencies between them. Over 60 % of BioInfer and GENIA complex interaction edges correspond to a single collapsed Stanford dependency. Longer paths are more common for the other parses. The paths for binary interactions are longer than for complex interactions.

have a corresponding dependency in the collapsed Stanford parse. With uncollapsed Stanford and Link Grammar parses, the shortest path more often consists of multiple edges. This supports the design choices of the collapsed Stanford scheme, which was developed to facilitate applications such as information extraction. It is very interesting that the complex interactions of both BioInfer and GENIA correspond so closely to this syntactic representation. The annotators of BioInfer and GENIA were biologists with no formal expertise on the syntactic structure of the sentences they were annotating. Yet the Stanford syntactic parse and the semantic annotations of BioInfer and GENIA, developed independently and with somewhat different aims, result in very similar graph structures.

For the BioInfer corpus, we also compared the complex interactions to the pairwise binary annotation for the same sentences. The shortest dependency paths corresponding to interaction edges were shorter for the complex interactions than the binary ones. In the case of the collapsed Stanford annotation, over 60 % of complex interaction edges linked neighbouring nodes in the dependency graph. For binary interactions the shortest path most commonly consisted of three dependencies.

For paths of length one, we also measured which dependency types correlated best with each

interaction graph edge type (Table 1). Certain edge types correspond very strongly to a specific dependency type. For example, an interaction edge of type *EQUAL* has most often a corresponding edge of type *appos* in the collapsed Stanford parse. This is promising for the development of systems for detection of interaction type.

[%]	appos	nn	nsubj	prep_of
EQUAL	**73.45**			
MEMBER	**27.27**	**59.60**		
agent	0.45	2.91	**22.6**	5.82
possessor		**31.96**		**48.45**
sub		**45.52**		2.76
super		**22.35**		**47.06**

Table 1: Selected BioInfer complex interaction edges (vertical) of which over 20 % have a one-to-one correlation to a collapsed Stanford format dependency (horizontal). The percentages are of all interaction edges, including those not corresponding to a single dependency. Values > 20 % are emphasized with bold text.

3 Conclusions and future work

Comparison of the interaction annotation to different parse schemes showed that the complex interactions of both BioInfer and GENIA are closer to the collapsed Stanford parse than to the other considered parse representations, supporting its value in extracting complex interactions.

The independently developed complex interaction formats of BioInfer and GENIA and the collapsed Stanford dependency parse are strikingly similar. We assume this indicates that these schemes succeed in capturing the essential structure and information of the annotated text. Our analysis is the first comparison of the relative complexity of BioInfer and GENIA interactions and our results suggest that they are of roughly similar complexity in this regard.

Comparison of complex and binary interactions indicates that while complex interactions can correspond closely to a syntactic dependency parse, binary interactions often link syntactically distant words. Therefore, despite appearances, complex interactions may prove to be easier to extract than binary ones. As previous studies have shown (Pyysalo et al., 2008), with binary interactions the definition of "interaction" also varies substantially, leading easily to ambiguous data. We hope that complex annotation will allow a more precise definition of the various concepts falling under the term "interaction", allowing both the development of better extraction systems and more consistent evaluation of the results.

These findings will be useful when we attempt to use the studied parses and annotations in the development of an automated system for the extraction of complex interactions. Our preliminary study indicates that the resources we evaluated can provide a consistent basis for future work.

Acknowledgments

This work was supported by the Academy of Finland.

References

A. Clegg and A. Shepherd. 2007. Benchmarking natural-language parsers for biological applications using dependency graphs. *BMC Bioinformatics*, 8(1):24.

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

J. Heimonen, S. Pyysalo, F. Ginter, and T. Salakoski. 2008. Complex-to-pairwise mapping of biological relationships using a semantic network representation. In *Proc. of SMBM'08*. To appear.

J-D. Kim, T. Ohta, and Tsujii J. 2008. Corpus annotation for mining biomedical events from literature. *BMC Bioinformatics*, 9(1):10.

Martin Krallinger, Florian Leitner, and Alfonso Valencia. 2007. Assessment of the second BioCreative PPI task: Automatic extraction of protein-protein interactions. In *Proc. of BioCreative II*, pages 41–54.

M.-C. de Marneffe, B. MacCartney, and C. D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proc. of LREC'06*, pages 449–454.

S. Pyysalo, F. Ginter, J. Heimonen, J. Björne, J. Boberg, J. Järvinen, and T. Salakoski. 2007. BioInfer: A corpus for information extraction in the biomedical domain. *BMC Bioinformatics*, 8(1):50.

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

D. D. Sleator and D. Temperley. 1991. Parsing English with a Link Grammar. Technical Report CMU-CS-91-196, Carnegie Mellon University.

Y. Tateisi, A. Yakushiji, T. Ohta, and J. Tsujii. 2005. Syntax annotation for the genia corpus. In *Proc. of the IJCNLP 2005*, pages 222–227.

Syntactic Pattern Matching with GraphSpider and MPL

Andrew B. Clegg
Research Dept. of Structural
and Molecular Biology
University College London
andrew.clegg@gmail.com

Adrian J. Shepherd
School of Crystallography
Birkbeck, University of London
a.shepherd@mail.cryst.bbk.ac.uk

Abstract

We present MPL (Metapattern Language), a new formalism for defining patterns over dependency-parsed text, and GraphSpider, a matching engine for extracting dependency subgraphs which match against MPL patterns. Using a regexp-like syntax, MPL allows the definition of subgraphs matching user-specified patterns which can be constrained by word or word class, part-of-speech tag, dependency type and direction, and presence of named variables in particular locations. Although MPL and GraphSpider are general-purpose, we developed a set of patterns to capture biomolecular interactions which achieved very high precision results (92.6% at 31.2% recall) on the LLL Challenge corpus. MPL specifications and pattern sets, and the GraphSpider software, are available on SourceForge: http://graphspider.sf.net/

1 Introduction

Various syntactic parsing methods have been used to provide input data for natural-language processing (NLP) tasks in the biomedical domain. The rich grammatical information provided by different kinds of parsers can be useful in relationship extraction (Riedel and Klein, 2005) and classification (Rosario and Hearst, 2004), event extraction (Yakushiji et al., 2001), semantic interpretation (Grover et al., 2005), named-entity recognition (Finkel et al., 2004), term extraction (Aronson, 2001), and information retrieval (Shi et al., 2005). This diversity of usage scenarios for syntax data, coupled with the growing availability of syntactically-annotated biomedical text (Tateisi et al., 2005; Kulick et al., 2004; Pyysalo et al., 2007a), suggests that general-purpose tools for querying and manipulating syntactic structures may be useful to researchers in this field.

In order to facilitate our experiments with dependency parsing of biomedical sentences, we developed a language for defining patterns over dependency graphs, plus a tool for matching patterns to sentences and extracting graph nodes of interest. Although this work was carried out in the context of a research project in gene/protein interaction extraction, there is nothing in either the language specification or software which is specific to this task or indeed to biological applications in general.

Several tools already exist for searching constituent tree representations of parsed sentences. The best-known of these is TGrep2[1] which is itself a successor to the original tgrep[2] that was developed alongside the Penn Treebank (Marcus et al., 1994). Both allow the construction of patterns of arbitrary complexity resembling hierarchical regular expressions, which constrain searches by words, part-of-speech (POS) tags and constituent labels, along with the positions in a subtree these elements must hold relative to each other. Similar features are provided by Tregex (Levy and Andrew, 2006). All of these tools were designed to operate on Penn Treebank-style trees; equivalents for other annotation schemas are provided by JAPE, part of the GATE package (Cunningham et al., 2007), CQP, part of the IMS Corpus Workbench (Christ, 1994), and Mother of Perl (Doran et al., 1996). NetGraph (Mírovský, 2008) is designed specifically for dependency graphs but is rather a complex client-server system. The OntoGene project at the University of Zurich has developed a system very similar to ours (Rinaldi et al., 2006), but it is available only via a web interface, with no source or binaries, making it much less useful for other researchers. None of these tools can perform noun phrase chunking on-the-fly (see below), and none provide native support for the Stanford dependency grammar (de Marneffe et al., 2006), which our own systems use, and which has been proposed as a convenient common schema for syntactic annotation and processing of biomedical text (Pyysalo et al., 2007b).

[1] http://tedlab.mit.edu/~dr/TGrep2/

[2] http://www.ldc.upenn.edu/ldc/online/treebank/

2 Metapattern Language (MPL)

An MPL file is composed of three kinds of rules, which taken together, specify a set of patterns to search for. **Match rules** define variables which hold either plain text strings or regular expressions, designed to match words, POS tags or dependency types in a graph. For example, the following rule declares a variable `@VERB` which matches any two or three-character string starting with the letters *VB*, and is designed to match POS tags like *VBN*, *VBZ* etc.:

```
match @VERB = ^VB.?$
```

Pattern rules are composed of variables, literal strings and connectors, and describe the subgraphs which the matching engine must attempt to find in the input sentences. Subgraphs are defined in terms of nodes (words with POS tags) connected by directional, labeled dependencies, although of course wildcard variables can be used in order to leave any of these elements unspecified. The simplest possible pattern simply matches a single node by tag and word, for example `NN~~regulation`. The following pattern matches fragments of the form ⟨*agent*⟩ *inhibits* ⟨*target*⟩; *inhibits* is specified literally, as are the POS tags (*VBZ* and *NN*) and the *dobj* direct object dependency, while the agent and target entities and the subject dependency refer to variables:

```
pattern
VBZ~~inhibits
   ( @NSUBJ NN~~@AGENT )
   ( dobj NN~~@TARGET )
end
```

Finally, **replacement rules** allow variations on explicitly-defined patterns to be generated automatically, to capture known wording alternatives or common variations on simple structures. They take the form of string replacement rules that are applied in turn to each of the patterns in the MPL file. They can operate on any part of a pattern rule (from a single word, POS tag or dependency to an entire subgraph) as long as the resulting pattern is well-formed. The following rule shows how a node matching a single string can be replaced by one matching a simple prepositional phrase:

```
replace @TARGET = expression
   ( prep_of NN~~@TARGET )
```

Applying this replacement rule to the example pattern given above results in the following pattern:

```
VBZ~~inhibits
   ( @NSUBJ NN~~@AGENT )
   ( dobj NN~~expression
     ( prep_of NN~~@TARGET ) )
```

This automatically-generated pattern will match sentence fragments like ⟨*agent*⟩ *inhibits expression of* ⟨*target*⟩. Note that since the pattern is defined over syntactic dependencies rather than linear strings of text, additional words intervening between any of the words covered by the pattern will not stop it matching, provided the grammatical relations between the matched words are correct. In other words, a sentence like ⟨*agent*⟩ *inhibits* ⟨*entity*⟩*-mediated expression of* ⟨*target*⟩ will still be matched.

Although one could attempt to define match rules to recognize named entities using regular expressions or lists of entity names, this is not likely to be successful except in very specific circumstances. Instead, we suggest that the user preprocesses the text with a named entity recognizer, then replaces all the entities found with placeholders (e.g. *Entityaa* ... *Entityzz*) that can be easily and unambiguously found by match rules using regular expressions. If a record of placeholder substitutions is kept, the original entity name behind each placeholder can be recovered trivially. Alternatively, if the variables such as `@AGENT` and `@TARGET` are defined with unrestricted wildcard expressions, then any word—or chunked phrase, see below—playing the appropriate syntactic role in a pattern will be identified. This approach turns the problem of named-entity recognition on its head, by assuming that *any* names found in expressions like *X inhibits expression of Y* represent biologically-interesting entities.

MPL offers several features beyond those described here, but does not yet support cycles or multiple parentage, meaning that its patterns are strictly trees rather than graphs. However, in evaluation (see below) we found no occasions when this was problematic.

3 GraphSpider

GraphSpider is a Java-based tool for performing MPL searches. It requires the Stanford parser distribution[3] to be installed, although it can accept the output of any constituent parser that uses

[3] `http://nlp.stanford.edu/software/lex-parser.shtml`

standard bracketed tree notation and Penn Treebank labels, or pre-generated Stanford-style dependency graphs in its own file format. If the text is supplied in trees, the conversion algorithm supplied with the Stanford parser is used to generate the dependency graphs (de Marneffe et al., 2006).

GraphSpider consists of two major components, an MPL parser and a matching engine. The MPL parser is responsible for reading in a pattern file supplied by the user, applying all replacement rules where possible in order to generate variant patterns, and compiling the patterns into in-memory representations. The matching engine then iterates over the sentences supplied, and for each one, finds every location where any pattern matches against the dependency graph of the sentence, including overlapping matches and locations where multiple patterns can match. It can then output the results for the sentence in one of several user-specified formats, ranging in scope from the entire sentence to just the nodes (words) that have matched against variables in the pattern.

Optionally, GraphSpider can apply a noun-phrase chunking algorithm to all constituent trees before converting them into dependency graphs. This simply identifies noun phrases with internal structure and flattens them into single words with the spaces replaced by underscores. The resulting graphs will tend to be much simpler, with single nodes encapsulating entire compound noun phrases (including adjectives, determiners and participles). However, MPL patterns must be written specifically to target chunked graphs, as pattern rules designed to match against traditional word-per-node graphs will not work on them. There is also a mechanism for plugging in Java classes for ad-hoc post-processing of the results, which we used to implement negation filtering.

4 Applications

Given a set of patterns capturing syntactic representations of biological events or interactions, and a corpus of parsed sentences, GraphSpider can be used to extract the entities, the keywords describing their relationships, and optionally any other words of interest. To a certain extent, phrasing variations and parse errors can be accounted for by the use of replacement rules to generate variant patterns automatically.

To test this approach, we developed a pattern set based on the training set from the LLL Challenge gene interaction task (Nédellec, 2005), and ran it against the test set, after replacing all gene/protein names in the sentences with placeholders. Although its coverage of the test set was comparatively low (31.2% recall), the predictions it did make were very accurate indeed (92.6% precision), suggesting that this method would be well-suited to unsupervised applications which require as little noise as possible in the results. We determined that these scores were achievable with as few as 29 hand-crafted patterns and 49 replacement rules, giving rise to 228 patterns in total (Clegg, 2008). Part of the reason for the low recall is that this method is sensitive to small parse errors which are not foreseen during the pattern engineering stage.

Another usage scenario is in exploratory corpus analysis and interactive text mining. By designing appropriate patterns, one can use GraphSpider to answer questions like "what entities bind to protein A?", "what temporal/locative modifiers are applied to expression of gene B?" (i.e. when/where does expression take place?), and "what verbs take a gene/protein phrase as their subject or object?". We have used this last technique to automatically extract keyword lists for the creation of further patterns.

The input/output and processing options supported by GraphSpider enable it to be used in a variety of alternative modes as well. For example, it can act simply as a noun phrase chunker, by bypassing the pattern matching engine completely in order to turn traditional constituent trees into chunked graphs. Similarly, it can strip the tree or graph annotation from a sentence and return just the plain text. And its ability to save and load dependency graphs in a human- and machine-readable format provides valuable functionality missing from the Stanford parsing toolkit.

We present MPL and GraphSpider in the hope that the NLP community finds them useful, and not just in the biological context where they were developed. All feedback, code or pattern contributions, and suggestions for future developments, are of course welcomed.

References

Alan R Aronson. 2001. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. In *Proceedings of the American Medical*

Informatics Association Symposium, pages 17–21. Hanley and Belfus, Inc.

Oliver Christ. 1994. A modular and flexible architecture for an integrated corpus query system. In *Proceedings of the Third Conference on Computational Lexicography and Text Research (COMPLEX '94)*, pages 23–32, Budapest.

Andrew B. Clegg. 2008. *Computational-Linguistic Approaches to Biological Text Mining*. PhD thesis, Birkbeck, London.

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Cristian Ursu, Marin Dimitrov, Mike Dowman, Niraj Aswani, Ian Roberts, Yaoyong Li, and Andrey Shafirin. 2007. *Developing Language Processing Components with GATE Version 4 (a User Guide)*. The University of Sheffield, `http://www.gate.ac.uk/`.

Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of 5th International Conference on Language Resources and Evaluation (LREC2006)*, Genoa, Italy, May.

Christine Doran, Michael Niv, Breck Baldwin, Jeffrey Reynar, and B. Srinivas. 1996. Mother of Perl: A multi-tier pattern description language. Technical report, Department of Computer Science, University of Pennsylvania.

Jenny Finkel, Shipra Dingare, Hoy Nguyen, Malvina Nissim, Christopher Manning, and Gail Sinclair. 2004. Exploiting context for biomedical entity recognition: From syntax to the web. In Nigel Collier, Patrick Ruch, and Adeline Nazarenko, editors, *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA)*, pages 88–91, Geneva, Switzerland, August 28–29.

Claire Grover, Mirella Lapata, and Alex Lascarides. 2005. A comparison of parsing technologies for the biomedical domain. *Natural Language Engineering*, 11(1):27–65.

Seth Kulick, Ann Bies, Mark Liberman, Mark Mandel, Ryan McDonald, Martha Palmer, Andrew Schein, Lyle Ungar, Scott Winters, and Pete White. 2004. Integrated annotation for biomedical information extraction. In Lynette Hirschman and James Pustejovsky, editors, *HLT-NAACL 2004 Workshop: BioLINK 2004, Linking Biological Literature, Ontologies and Databases*, pages 61–68, Boston, Massachusetts, USA, May 6. Association for Computational Linguistics.

Roger Levy and Galen Andrew. 2006. Tregex and Tsurgeon: tools for querying and manipulating tree data structures. In *Proceedings of 5th International Conference on Language Resources and Evaluation (LREC2006)*, Genoa, Italy, May.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1994. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Jiří Mírovský. 2008. Netgraph—making searching in treebanks easy. In *Proceedings of the Third International Joint Conference on Natural Language Processing (IJCNLP 2008)*, pages 945–950, Hyderabad, India, January.

Claire Nédellec. 2005. Learning Language in Logic—Genic Interaction Extraction Challenge. In *Proceedings of Learning Language in Logic (LLL05)*, Bonn, Germany, August.

Sampo Pyysalo, Filip Ginter, Juho Heimonen, Jari Björne, Jorma Boberg, Jouni Järvinen, and Tapio Salakoski. 2007a. BioInfer: a corpus for information extraction in the biomedical domain. *BMC Bioinformatics*, 8(50), February.

Sampo Pyysalo, Filip Ginter, Veronika Laippala, Katri Haverinen, Juho Heimonen, and Tapio Salakoski. 2007b. On the unification of syntactic annotations under the Stanford dependency scheme: A case study on BioInfer and GENIA. In *Proceedings of the Workshop on BioNLP 2007: Biological, translational, and clinical language processing*, pages 25–32, Prague, Czech Republic, June. Association for Computational Linguistics.

Sebastian Riedel and Ewan Klein. 2005. Genic interaction extraction with semantic and syntactic chains. In *Proceedings of Learning Language in Logic (LLL05)*, Bonn, Germany, August.

Fabio Rinaldi, Gerold Schneider, Kaarel Kaljurand, Michael Hess, and Martin Romacker. 2006. An environment for relation mining over richly annotated corpora: the case of GENIA. *BMC Bioinformatics*, 7 (Suppl 3)(S3), November.

Barbara Rosario and Marti Hearst. 2004. Classifying semantic relations in bioscience texts. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, pages 430–437, Barcelona, Spain, July.

Zhongmin Shi, Baohua Gu, Fred Popowich, and Anoop Sarkar. 2005. Synonym-based query expansion and boosting-based re-ranking: A two-phase approach for genomic information retrieval. In *Proceedings of the Fourteenth Text REtrieval Conference (TREC 2005)*, Gaithersburg, Maryland, November.

Yuka Tateisi, Akane Yakushiji, Tomoko Ohta, and Jun'ichi Tsujii. 2005. Syntax annotation for the GENIA corpus. In *Proceedings of the International Joint Conference on Natural Language Processing 2005, Companion volume*, pages 222–227, Jeju Island, Korea, October.

Akane Yakushiji, Yuka Tateisi, Yusuke Miyao, and Jun'ichi Tsujii. 2001. Event extraction from biomedical papers using a full parser. In *Proceedings of the Sixth Pacific Symposium on Biocomputing*, pages 384–395. World Scientific Publishing.

Accurate Conversion of Dependency Parses: Targeting the Stanford Scheme

Katri Haverinen,[1] **Filip Ginter,**[1] **Sampo Pyysalo**[2] and **Tapio Salakoski**[1,2]
[1]Department of Information Technology
[2]Turku Centre for Computer Science (TUCS)
20014 University of Turku, Finland
first.last@utu.fi

Abstract

We present a conversion from the dependency scheme employed by the Pro3Gres parser to the Stanford scheme, as a further step towards unification of dependency schemes. An evaluation of the conversion shows that it is highly reliable, resulting in less than one percentage point performance penalty on the actual parser output. This supports the suitability of the Stanford scheme as a unifying representation and the applicability of our conversion formalism to parser scheme conversions. We further provide an evaluation of the Pro3Gres parser, thus adding it to the growing set of parsers evaluated under comparable conditions using the Stanford scheme.

1 Introduction

The development of parsing technologies has recently made it feasible to apply full parsers to many tasks where partial parsing was previously the approach of choice, such as information extraction (IE). In particular in biomedical IE, there has been substantial interest in the application of full dependency parsers in response to the relative complexity of the domain language and also due to the advantages of the immediate representation that dependency formalisms give to grammatical functions (e.g. *subject* and *object*).

Parsing technologies, however, differ substantially in the syntactic schemes employed. This has a number of unfortunate consequences: corpora tend to be formalism-specific, reducing the amount of data available, evaluations of parsers yield results that cannot be directly compared, and methods that apply parsers tend to become bound to a particular scheme. Both parser developers and those who apply parsers would benefit from a reduction of this fragmentation.

In this study, we consider a full dependency parser, Pro3Gres (Schneider et al., 2004), which has been developed with particular attention to the challenges of biomedical domain text and applied in numerous domain studies. Pro3Gres has been evaluated by its authors on a small dependency treebank in its native syntactic representation as well as in one of the CoNLL shared tasks on dependency parsing (Schneider et al., 2007); however, due to differences in syntactic representations it is difficult to directly relate these results to evaluations of other parsers in the domain. Here, we study the feasibility of translating the unique syntactic scheme of Pro3Gres into a more commonly used shared representation.

2 Related work

There has recently been a significant amount of work narrowing the gap between different parser output representations. Three prominent approaches are dependency-based: the Grammatical Relations (GR) dependency scheme, proposed by Carroll et al. (1998) for parser evaluation, the Stanford dependency scheme (SD) of de Marneffe et al. (2006), oriented towards applications such as IE, and the scheme that was introduced in the CoNLL shared dependency parsing tasks (Nivre et al., 2007). In this paper, we consider unification under the Stanford scheme.

The GR and SD schemes have been applied in a number of parser evaluation studies in which the native parser output was converted into the target dependency scheme. Table 1 summarizes estimated performance of the various conversions as

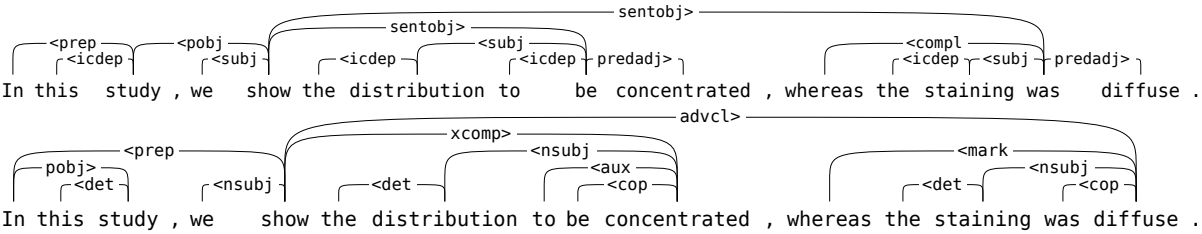


Figure 1: An example of the differences between the Pro3Gres scheme (top) and the Stanford scheme (bottom). Note the technical intra-chunk dependencies, *icdep*, in the Pro3Gres parse.

study	from	to	F
Clark and Curran (2007)	CCG	GR	84.8%
Pyysalo et al. (2007b)	LG	SD	97.1%
Sagae et al. (2008)	HPSG	GR	87.1%
Sagae et al. (2008)	SD	GR	74.5%
Sagae et al. (2008)	HPSG	PTB	98.1%

Table 1: Previously reported conversions with conversion quality estimates, given as F-scores.

reported by their authors.

There is a surprisingly large amount of variation in these results. While the results would appear to suggest that conversions into GR are particularly difficult, there are differences in conversion methodology that prevent clear conclusions from being drawn. Additionally, the schemes are different in the sense that some of them, including GR, are deep, whereas others are more surface-oriented. The development and evaluation of a conversion from the Pro3Gres native scheme to SD is thus an important point towards establishing whether highly accurate conversions into SD can be achieved in general.

3 Methods

We now briefly describe the Pro3Gres and SD schemes and the Pro3Gres→SD conversion. For details of the two schemes, see the papers by Schneider et al. (2004) and de Marneffe et al. (2006), respectively.

3.1 Pro3Gres parser and its dependency scheme

Pro3Gres is a dependency-based parser created by Schneider et al. (2004). A notable property of the parser is that it uses a chunker to extract noun and verb groups as a separate pre-parsing step.

The Pro3Gres scheme has a total of 23 dependency types, excluding the so called *intra-chunk dependencies* that are fully contained within chunks. As intra-chunk dependencies are not a primary output of the parser, and as they form a relatively flat structure, our conversion does not target them. However, in order to be able to recognize certain structures, such as passives, we introduce technical dependencies *icdep* from the chunk head to each token in the chunk. Figure 1 is an illustration of the Pro3Gres scheme as compared to the Stanford scheme.

3.2 Stanford dependency scheme

The Stanford dependency scheme (SD) is an application-oriented scheme introduced by de Marneffe et al. (2006). The scheme defines 48 dependency types that are arranged in a hierarchy. De Marneffe et al. also provide a method for converting parse trees from the PTB scheme into the SD scheme.

3.3 Pro3Gres→SD conversion

The Pro3Gres→SD conversion was carried out using 176 hand-written rules in the lp2lp dependency parse conversion formalism (see, e.g., Pyysalo et al. (2007b)).

One-to-one correspondences of dependency types are rare in the conversion. An example of a particularly difficult dependency type to translate is the Pro3Gres type *sentobj*. In SD, it corresponds to five different dependency types: *xcomp*, *partmod*, *infmod*, *ccomp* and *advcl*. In Figure 1 we illustrate two different uses of the *sentobj* type. Another issue that complicates the transformation rules is that some dependency types in SD, the most common example being the copula, cause substantial changes to the structure of the parse, as the head is chosen differently in the two schemes. This is, again, illustrated in Figure 1.

4 Results and discussion

We estimate the conversion performance in two separate ways: on an actual output of the

		gold standard	
		present	*absent*
system output	*present*	461	77 (73+4)
	absent	161 (156+5)	—

Table 2: Results of the manual analysis of the conversion quality. Parsing errors are divided between errors attributed to the Pro3Gres parser and errors attributed to the conversion. This division is shown in parentheses as *parser errors+conversion errors*. All numbers are dependency counts.

err.	P	R	F
incl.	85.7% (461/538)	74.1% (461/622)	79.5%
excl.	86.3% (461/534)	74.9% (466/622)	80.2%

Table 3: (P)recision, (R)ecall and F-score figures including and excluding conversion errors (based on the manual analysis reported in Table 2).

Pro3Gres parser and on a separate set of gold-standard Pro3Gres parses. The former evaluation is performed on the BioInfer corpus (Pyysalo et al., 2007a) which has gold-standard SD annotation. As performance measures, we use precision, recall, and F. The rules have been developed using 200 sentences from BioInfer as reference, we thus perform all BioInfer measurements on an evaluation set consisting of the remaining 900 sentences.

4.1 Evaluation of the Pro3Gres→SD conversion

To estimate the quality of the conversion, we manually analyse the converted Pro3Gres output on 30 sentences (622 dependencies) randomly drawn from the evaluation set of BioInfer sentences. We attribute each parsing error as caused either by the parser or by the conversion. The result of this analysis is presented in Table 2. We find that the conversion accounts for 4/77=5.2% of all precision errors and 5/161=3.1% of all recall errors. The conversion thus accounts for only a small percentage of the errors found in the converted parser output. In fact, the absolute penalty on the overall F-score of the parser is only 0.7 percentage points, as shown in Table 3.

The manual analysis estimates the performance of the rules on the actual parser output and is thus most relevant from the applied point of view and for parser evaluation. As seen in Table 3, Pro3Gres trades higher precision for lower recall. This often means that rare and exceptionally complex structures are not given any analysis. This, in turn, has the effect that also the conversion rules are not applied for these sections of the sentence and therefore cannot fail. In order to estimate the performance of the conversion in the ideal case of the parser producing a perfect analysis, we have annotated in both the Pro3Gres scheme and the SD scheme a set of 50 sentences (715 SD dependencies) randomly drawn from the GENIA corpus. On this set, we find that the conversion results in a 96.1% F-score (96.9% precision and 95.4% recall). The difference in conversion accuracy of the actual parser output as compared to the gold-standard output shows that as the parser coverage is increased in the future, corresponding conversion rules will need to be added.

4.2 Evaluation of the Pro3Gres parser

The Pro3Gres→SD conversion allows an evaluation of Pro3Gres performance on the SD-annotated BioInfer corpus, thus complementing the results previously reported by Clegg and Shepherd (2007) and Pyysalo et al. (2007b). This evaluation, however, is complicated by the fact that Pro3Gres chunks noun and verb groups and does not aim to generate sufficiently detailed chunk-internal analysis. To address this difference in resolution detail, we chunk the gold-standard data using the existing gold-standard annotation and only consider chunk-external dependencies in the evaluation (see Figure 2).

In Table 4 we report the performance of Pro3Gres on the 900 BioInfer evaluation sentences. The parser was used together with the GENIA tagger (Tsuruoka et al., 2005) and LTChunk chunker (Mikheev, 1997). As a point of comparison, we also report the performance of the Charniak-Lease parser (Lease and Charniak, 2005), a state-of-the-art, domain-adapted statistical parser. The Charniak-Lease output was transformed to the SD scheme using the Stanford conversion tools (de Marneffe et al., 2006). To assess the numerical comparability of the chunk-based evaluation strategy, we include the result reported by Pyysalo et al. (2007b) for the Charniak-Lease parser on full, unchunked BioInfer.

We observe that Pro3Gres achieves state-of-the-art performance, only slightly lower than that of the Charniak-Lease parser. Further, we note that the chunked evaluation strategy results in 3.5 percentage point performance penalty.

	Pro3Gres			Charniak-Lease			
chunked	P	R	F	P	R	F	ΔF
yes	78.5	70.5	74.3	74.4	77.5	75.9	1.6
no	-	-	-	78.4	79.9	79.4	-

Table 4: Performance of the Pro3Gres and Charniak-Lease parsers on the BioInfer corpus. The result for the Charniak-Lease parser on the unchunked BioInfer was reported by Pyysalo et al. (2007b).

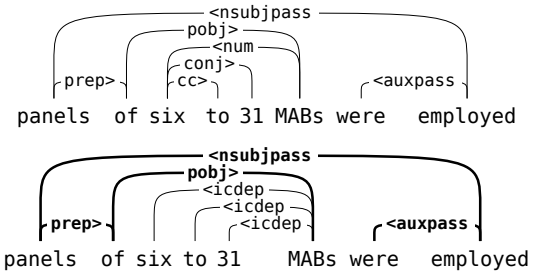


Figure 2: Original gold-standard structure (top) compared to chunked gold standard (bottom) with the intra-chunk structure flattened into *icdep* dependencies. The parser is only evaluated on the chunk-external dependencies, displayed in bold.

5 Conclusions

The main practical contribution of this paper is the set of rules for a very accurate conversion from the Pro3Gres scheme to the Stanford scheme (SD). In particular, on actual parser output, the conversion results in less than one percentage point penalty on the parser F-score performance. The conversion increases the applicability of Pro3Gres, as it enables it to produce output in a commonly used scheme.

Moreover, the ability to produce an accurate conversion into the SD scheme, already a third such conversion — the other two being the conversions from PTB (de Marneffe et al., 2006) and from LG (Pyysalo et al., 2007b) — suggests that the SD scheme does not pose significant problems as a conversion target. The SD scheme is also designed to be oriented towards applications, such as IE (de Marneffe et al., 2006). This study thus further strengthens the case for the adoption of the SD scheme as a unifying representation for full parsers in the applied domain, previously argued for by de Marneffe et al. (2006), Clegg and Shepherd (2007), and Pyysalo et al. (2007b).

The evaluation data, the conversion rules, and our modified version of the lp2lp implementation are available under an open-source license at `http://www.it.utu.fi/BioInfer`.

6 Acknowledgments

We thank Gerold Schneider for producing the Pro3Gres parses. This study was supported by the Academy of Finland.

References

J. E. Carroll, E. Briscoe, and A. Sanfilippo. 1998. Parser evaluation: a survey and a new proposal. In *Proc. of LREC'98*, pages 447–454.

S. Clark and J. Curran. 2007. Formalism-independent parser evaluation with CCG and DepBank. In *Proc. of ACL'07*, pages 248–255.

A. B. Clegg and A. Shepherd. 2007. Benchmarking natural-language parsers for biological applications using dependency graphs. *BMC Bioinformatics*, 8(1):24.

M. Lease and E. Charniak. 2005. Parsing biomedical literature. In *Proc. of IJCNLP'05*, pages 58–69.

M-C. de Marneffe, B. MacCartney, and C. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proc. of LREC'06*, pages 449–454.

A. Mikheev. 1997. Automatic rule induction for unknown-word guessing. *Computational Linguistics*, 23(3):405–423.

J. Nivre, J. Hall, S. Kübler, R. McDonald, J. Nilsson, S. Riedel, and D. Yuret. 2007. The CoNLL 2007 shared task on dependency parsing. In *Proc. of EMNLP-CoNLL'07*, pages 915–932.

S. Pyysalo, F. Ginter, J. Heimonen, J. Björne, J. Boberg, J. Järvinen, and T. Salakoski. 2007a. BioInfer: A corpus for information extraction in the biomedical domain. *BMC Bioinformatics*, 8(50).

S. Pyysalo, F. Ginter, V. Laippala, K. Haverinen, J. Heimonen, and T. Salakoski. 2007b. On the unification of syntactic annotations under the Stanford dependency scheme: A case study on BioInfer and GENIA. In *Proc. of BioNLP'07*, pages 25–32.

K. Sagae, Y. Miyao, T. Matsuzaki, and J. Tsujii. 2008. Challenges in mapping of syntactic representations for framework-independent parser evaluation. In *Proc. of ICGL'08*.

G. Schneider, F. Rinaldi, and J. Dowdall. 2004. Fast, deep-linguistic statistical dependency parsing. In *Proc. of COLING'04 Recent Advances in Dependency Grammar*, pages 33–40.

G. Schneider, K. Kaljurand, F. Rinaldi, and T. Kuhn. 2007. Pro3Gres parser in the CoNLL domain adaptation shared task. In *Proc. of EMNLP-CoNLL'07*, pages 1161–1165.

Y. Tsuruoka, Y. Tateishi, J-D. Kim, T. Ohta, J. McNaught, S. Ananiadou, and J. Tsujii. 2005. Developing a robust part-of-speech tagger for biomedical text. In *Proc. of PCI'05*, pages 382–392.

Classifying Verbs in Biomedical Text Using Subject-Verb-Object Relationships

Pieter van der Horn and **Bart Bakker** and **Gijs Geleijnse** and **Jan Korst**
Philips Research Laboratories
High Tech Campus 12a, 5656 AE Eindhoven, The Netherlands
{pieter.van.der.horn,bart.bakker,gijs.geleijnse,jan.korst}@philips.com

Sergei Kurkin
BioFocus DPI, a Galápagos company
Darwinweg 24, P.O. Box 127, 2300 AC Leiden, The Netherlands
sergei.kurkin@glpg.com

Abstract

A protein-protein interaction in a biomedical text is often described using a wide range of verbs, e.g. *activate, bind, interact*. In order to determine the specific type of interaction described, we must first determine the meaning of the verb used. In biomedical context, however, some verbs can be considered synonyms, yet may not be so in standard lexical databases, like WordNet. Furthermore, some verbs will not be mentioned at all in such a dictionary, since they are too area specific. We propose a simple classification scheme to predict the correct class (meaning) of the verb. With this, one can identify the types of protein-protein interactions described in subject-verb-object constructions in PubMed abstracts.

1 Introduction

Since scientific journals are still the most important means of documenting biological findings, biomedical articles are the best source of information we have on protein-protein interactions. The mining of this information will provide us with specific knowledge of the presence and types of interactions, and the circumstances in which they occur.

There are various linguistic constructions that can describe a protein-protein interaction. (Tateisi et al., 2004) use predicate-argument structures in the mining of protein-protein interactions. These are used to identify the specific roles of encountered proteins in an interaction, but not to determine the biomedical meaning of the verb itself. In (Wattarujeekrit et al., 2004), an extended model based on PropBank (Palmer et al., 2005) is used to group verbs according to their differences and similarities in sense, structure and number of arguments between their use in biomedical and regular text. Their main focus is domain-specific verbs that are used to describe molecular events in biology. We share their considerations about the fact that verbs are used in a different way in biomedical text compared to regular text. Our goal, however, is to group general verbs according to their meanings in biomedical text. This will allow us to identify the type of interaction indicated by any possible verb encountered, instead of having to rely on a limited number of predefined domain-specific verbs. Following (Jensen et al., 2006), we focus on causality to create a biologically meaningful distinction. We use two classes of verbs, making the distinction between relations that describe proteins *affecting* other proteins (*causal relation*) and any other relation (*non-causal relation*). Future work will incorporate more classes in order to be able to make a more specific distinction between different meanings of verbs.

2 Preprocessing

The protein-protein interactions we are interested in are described in the subject, the object and the interlinking verb phrase of a sentence. To determine which parts of the sentence make up this construction, we need to preprocess the sentence. For this, we use the Genia Chunker[1] to break the sentence into different chunks (in particular we are interested in noun phrases and verb phrases). We combine this information with the result of the Stanford Dependency Parser[2] to determine how these different chunks (phrases) are connected to

[1] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/tagger/
[2] http://nlp.stanford.edu/downloads/lex-parser.shtml

each other.

Using WordNet (Fellbaum, 1998), we can increase the number of verbs for which we know (or can reasonably assume) the right class. WordNet identifies synonyms for each verb, grouped by different senses (meanings) that are ordered by frequency (most common meaning first). We can choose how many senses we use (at least one), and how many recursive levels of synonyms we want (synonyms, synonyms of those synonyms, etc). However, this can create noise, since WordNet is a lexicon for the general use of words, and not specifically for biomedical context (Poprat et al., 2008). Lacking a proper biomedical lexicon, we will make limited use of WordNet in order to test our approach.

3 Classification

The subject-verb-object construction can be schematically represented as follows:

[(state of) protein] [verb] [(state of) protein]

We assume the interaction between the two proteins to be determined by the combination of the states in the noun phrases and the relevant verb in the verb phrase. Such states can be described by single words (e.g. *activation*, *suppression*, *overexpression*) or far more complicated descriptions. However, detection of these descriptions of states of proteins can be difficult and is a separate research topic. Since the focus of this paper is on the meanings of verbs, we will leave this detection of protein states for future work.

We make a distinction between two classes of verbs. One class describes a strict *causal relation* and the other covers all other types of meanings (*non-causal*). Table 1 shows some example verbs for the two classes.

Class	**Examples**
causal	*activate, inhibit, cause*
non-causal	*interact, require, bind*

Table 1: Two classes of verbs.

The second class includes not just verbs that describe a correlation (e.g. *interact*), but also verbs such as *require* and *bind*. One could argue that these latter verbs also describe a directed action from agent to target, like a strict causal relation does. However, they do not describe a direct change of the state of the target protein, and therefore we choose not to put them in the first class. The three verbs in the *causal* class represent the positive, negative and general causal relations. The three verbs in the *non-causal* class represent three different types of relations that occur very often in the text. Since these relations are not synonymous to each other, each of them has to be represented by a separate verb. Having labeled these six verbs manually, we will use this to attempt to automatically predict the right class for the possibly many unknown verbs that can occur in subject-verb-object constructions in biomedical text.

3.1 Naive Bayesian Classifier

Using a Naive Bayesian Classifier, we estimate the probability that a given verb belongs to a certain class. Bayes' Theorem describes this probability.

$$P(c_i|V) = \frac{P(c_i) \cdot P(V|c_i)}{P(V)} \quad (1)$$

In the retrieved subject-verb-object constructions, a verb V will occur a number of times, each time in combination with a specific ordered pair of proteins pp_j, one in the subject and one in the object.

$$V \equiv \{pp_1, pp_2, ..., pp_n\}$$

These pairs of proteins are the different features of this verb. In Naive Bayesian Classification, these features are assumed to *independently* contribute to the estimate of the posterior probability.

$$\begin{aligned} P(c_i|V) &= P(c_i|pp_1, pp_2, ..., pp_n) \\ &= \frac{P(c_i) \cdot \prod_{j=1}^{n} P(pp_j|c_i)}{P(pp_1, pp_2, ..., pp_n)} \end{aligned} \quad (2)$$

Given a test set of instances (in Section 4 we elaborate on how we get those instances), we define the following variables:

$f_{j,i}$	*number of occurrences of protein pair pp_j around verbs of class c_i (frequency)*
$Q_i = \sum_j f_{j,i}$	*number of protein pairs around verbs of class c_i*
$Q = \sum_i Q_i$	*total number of protein pairs encountered*
U	*number of unique protein pairs encountered in the training set*

With these, we can estimate the necessary probabilities:

$P(c_i) \cong \frac{Q_i}{Q}$ *prior probability of class* c_i

$P(pp_j|c_i) \cong \frac{f_{j,i}+1}{Q_i+U}$ *conditional probability of pair* pp_j *given class* c_i

For the conditional probability, we use Laplace estimates. That is, we add 1 to the numerator and U to the denominator, in order to compensate for pairs for which $f_{j,i} = 0$. If we would use $P(pp_j|c_i) \cong \frac{f_{j,i}}{Q_i}$ instead, the conditional probability would become equal to 0 if $f_{j,i} = 0$. This would cause the posterior probability $P(c_i|pp_1, pp_2, ..., pp_n)$ to be equal to 0 as well (Equation 2), leaving us without a reasonable estimate of this posterior probability. The probability $P(V)$ is the factor with which we normalize the numerator of Equation 2 for each class c_i. This gives us $P(c_i|V)$ for each class. A verb V is then classified to be in the class c_i for which the posterior probability $P(c_i|V)$ is highest.

$$C(V) = \underset{c_i}{\operatorname{argmax}}\, P(c_i|V)$$

4 Experiments

4.1 Setup

In order to test our approach, we retrieved a set of subject-verb-object relations from abstracts stored in PubMed. We chose to test our approach on yeast proteins rather than e.g. human proteins to avoid Named Entity Recognition problems. We used a predefined data set of names to detect yeast proteins in text.

To remove any excess information, the verb phrases are normalized. We assume the last verb in the phrase to be the relevant verb and check the direction of the relation (active or passive form of that verb). Finally, the verb is stemmed using the Porter stemmer (Porter, 1980). For those verbs that are in the passive form, the order of the protein pairs around it was reversed, and, for simplification, verb phrases that describe a negation were removed. More than one protein can occur in the subject and/or object, so we count each possible pair as an occurrence around the particular verb.

We used the 6 verbs as shown in Table 1 as a starting set to test the classifier. The training set is then augmented using WordNet. For the resulting verbs in the classes, we run a leave-one-out cross validation. That means, we classify each of these verbs by training the Naive Bayesian Classifier on the frequencies of the occurring pairs of proteins around the other known verbs. Some verbs we retrieved from WordNet may not occur at all in the subject-verb-object instances we have. These verbs are ignored in the leave-one-out cross-validation.

4.2 Results

	V	*C*	*A*	*P*
no WN	6	3	0.50	0.66
l1/s1	13	7	0.54	0.50
l1/s2	18	13	0.72	0.05
l1/sa	19	14	0.74	0.03
l2/s1	19	12	0.63	0.18
l2/s2	27	21	0.78	2.96E-3
l2/sa	55	32	0.58	0.14
l3/s1	26	20	0.77	4.68E-3
l3/s2	42	35	0.83	7.55E-6
l3/sa	73	43	0.59	0.08

Table 2: Leave-one-out cross-validation results.

Table 2 shows the results of the different tests, using different parameter settings in WordNet to augment the training set. They contain the number of verbs classified in the leave-one-out cross-validation (*V*), the number of verbs that were correctly classified (*C*), the accuracy ($A = \frac{C}{V}$) and the probability P that a random classifier would perform as good or better than this classifier, given by

$$P = \sum_{i=C}^{V} \binom{V}{i} p^i \cdot (1-p)^{V-i}$$

in which $p = \frac{1}{2}$ (determined by the number of classes). We have run the program with different settings for WordNet ('l1' means recursive level 1, 's2' means WordNet senses 1 to 2, 'sa' means all WordNet senses are taken).

From the cross-validations, we can see that the algorithm performs reasonably well. There are multiple settings that obtain an accuracy higher than 0.70, and one setting in particular ('l3/s2') reached an accuracy of 0.83. The probability that a random classifier would perform as good or better than this is $7.55 \cdot 10^{-6}$. In Table 3, the results of the cross-validation for this setting are shown for each of the 42 verbs, highest P_1 first (P_1 is

verb	P_1	error	verb	P_1	error
suppress	1.00	0	turn	0.68	0
have	1.00	0	position	0.67	1
lead	1.00	0	perform	0.59	0
activate	1.00	0	make	0.52	0
stimulate	1.00	0	comprise	0.51	0
reduce	1.00	0	investigate	0.49	0
cause	1.00	0	see	0.49	0
contain	1.00	0	incorporate	0.49	1
induce	1.00	0	do	0.49	1
repress	1.00	0	impact	0.49	0
allow	0.99	0	pull	0.49	0
control	0.99	0	situate	0.49	0
inhibit	0.97	0	displace	0.49	1
trigger	0.95	0	hold	0.49	1
give	0.85	0	attach	0.35	0
carry	0.81	0	occupy	0.32	0
keep	0.80	0	need	0.18	0
expect	0.79	1	involve	0.06	0
maintain	0.78	0	bind	2.64E-15	0
bear	0.75	0	interact	5.74E-30	0
affect	0.75	1	require	1.12E-54	0

Table 3: Cross-validation of 42 verbs.

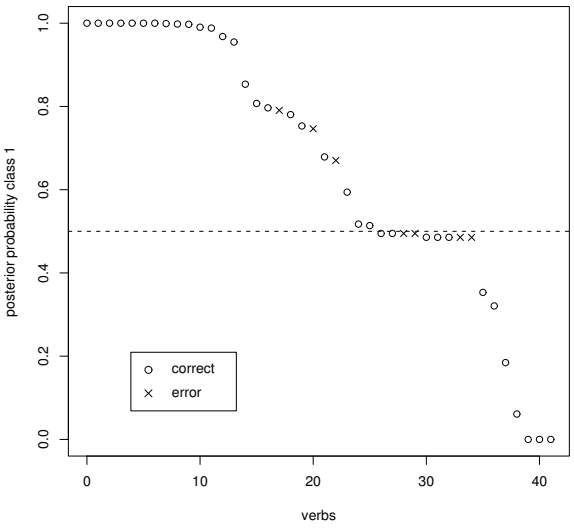


Figure 1: Results of leave-one-out cross-validation.

the posterior probability that a verb belongs to class 1, the *causal* class). Figure 1 visualizes the distances of the classifications from the decision boundary. The crosses indicate the errors made. The confidence of the classification is defined by distance of the posterior probability to the decision boundary. This confidence clearly differs for each verb. We can see that there is a group of 11 verbs for which the confidence is very low (*make* to *hold*). This group accounts for four of the errors made, out of a total of seven. For these 11 verbs it is unclear, even for humans, which class they belong to. Some of these verbs, however, may not describe any interaction at all. One could use a confidence threshold to discard the verbs of which the classifications are very uncertain.

5 Conclusions and Future Work

Given an appropriate set of known verbs, we can predict the meanings of unknown verbs with reasonable confidence. This automatic prediction is very useful, since it is infeasible to manually determine the meanings of all possible verbs. We chose to use a two-way distinction as a first step. Verbs like *require* and *bind* describe biologically distinct interactions however, and preferably should be put into classes separate from general correlations. In order to create a more detailed network of interacting proteins, one can take these other types into account as well.

Furthermore, it would be useful to separate the causal relationship into positive and negative relations. This specific distinction however is not just described in the connecting verb, but also in possible state descriptions in the noun phrases. Further research is necessary to extract these descriptions from the text. Finally, it would be useful to look at different syntactic constructions, other than just subject and object.

References

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press, May.

Lars J. Jensen, Jasmin Saric, and Peer Bork. 2006. Literature mining for the biologist: from information retrieval to biological discovery. *Nature Reviews Genetics*, 7(2):119–129.

M. Palmer, D. Gildea, and P. Kingsbury. 2005. The proposition bank: an annotated corpus of semantic roles. *Computational Linguistics*, (31):71–105.

M. Poprat, E. Beisswanger, and U. Hahn. 2008. Building a biowordnet using wordnet data structures and wordnet's software infrastructure - a failure story. In *ACL 2008 workshop "Software Engineering, Testing, and Quality Assurance for Natural Language Processing"*.

M.F. Porter. 1980. An algorithm for suffix stripping. In *Program*, volume 14, pages 130–137.

Y. Tateisi, T. Ohta, and J. Tsujii. 2004. Annotation of predicate-argument structure on molecular biology text. In *IJCNLP-04*.

T. Wattarujeekrit, P. K. Shah, and N. Collier. 2004. Pasbio: predicate-argument structures for event extraction in molecular biology. *BMC Bioinformatics*, 5, October.

Protein Name Tagging in the Immunological Domain

Renata Kabiljo
School of Crystallography Birkbeck
University of London
Malet Street, London WC1E 7HX
r.kabiljo@mail.cryst.bbk.ac.uk

Adrian J Shepherd
School of Crystallography Birkbeck
University of London
Malet Street, London WC1E 7HX
a.shepherd@mail.cryst.bbk.ac.uk

Abstract

The research described in this paper addresses the following question: How well do generic protein/gene name taggers perform when they are applied to full-text articles from the sub-domain of immunology (a sub-domain with its own distinctive protein nomenclature)? To answer this question we have created a new corpus – ImmunoTome – consisting of ten full-text immunological articles in which the names of proteins have been manually annotated. Our results show that a single tagger – ABNER trained on the BioCreAtivE corpus – performs significantly better than the other taggers we evaluated when applied to ImmunoTome. ImmunoTome is available from immunominer.cryst.bbk.ac.uk/tome.html.

1 Introduction

Large amounts of useful immunological data are to be found exclusively in full-text journal articles. Much of these data concern the key proteins involved in the immune response, notably antibodies, antigenic proteins, and cytokines. For our research as members of the ImmunoGrid Consortium[1], our ultimate aim is to devise methods capable of automatically extracting this information from the literature. As a crucial first step, we need to identify the protein entities themselves.

Some of the key protein entities of the immune system (or the genes that encode them) have their own, distinctive nomenclature, notably the CD nomenclature for leukocyte surface molecules (e.g. *CD4*, *CD8*) and the HLA nomenclature for the human leukocyte antigen system (e.g. *B40*, *DR14*, *Dw25*). Other important classes of immune system proteins have names that typically start with three upper-case letters, e.g. *TCR* (T cell receptors). Names containing a mixture of upper-case letters and digits are likely to be particularly easy for taggers to identify, as relatively few non-protein words have this form.

There are a number of freely-available named-entity taggers for proteins and other biomedical entities. These taggers have typically been trained on one or more of three widely-used biomedical corpora – GENIA, BioCreAtivE and Yapex. All three corpora consist of manually annotated abstracts (or sentences from abstracts) taken almost exclusively from non-immunological papers.

This raises an important question: How well do taggers trained on such corpora perform when they are applied to a specific sub-domain such as immunology characterized by its own distinctive protein nomenclature? This question is the starting point for the research described in this paper. Here we compare the performance of four freely-available taggers designed to annotate the names of proteins and other biomedical entities in natural language texts. The four taggers are: LingPipe[2], trained on GENIA (Kim *et al.*, 2003); NLProt (Mika & Rost, 2004), trained on Yapex (Franzén *et al.*, 2002); Gapscore (Chang *et al.*, 2004), a rule-based tagger; and ABNER (Settles, 2005). ABNER can be run in two modes: one trained on a simplified version of GENIA known as the JNLPBA corpus (Kim *et al.*, 2004), and the other trained on BioCreAtivE (Yeh *et al.*, 2005).

2 Methods

2.1 The ImmunoTome corpus

In order to assess the performance of generic protein taggers on full-text immunological articles, we created a new corpus – ImmunoTome. ImmunoTome consists of ten full-text articles from the *Journal of Immunology*, each containing at least one reference to the proteins *CD4* or *CD8*. (The latter criterion was adopted because we are particularly interested in the molecular aspects of the adaptive immune system.)

[1] www.immunogrid.eu

[2] www.alias-i.com/lingpipe/index.html

In ImmunoTome, protein names are annotated regardless of their context. For example, in the phrase "CD4+CD8- cells" both "CD4" and "CD8" are annotated as proteins. We have annotated both the names of proteins and the names of the genes that code for those proteins, but not non-coding entities such as promoters and enhancers. We believe this approach is a reasonable compromise for many biomedical text-mining applications, as a clear distinction between protein and gene names is often impossible.

In designing ImmunoTome, we have aimed to adopt good practices relevant to the development of biomedical corpora, including the provision of explicit annotation guidelines. ImmunoTome was created by two annotators with prior experience of developing the ProSpecTome corpus (Kabiljo *et al*., 2007). Inter-annotator agreement was calculated using a single article after an iterative process of guideline and annotation refinement using the other nine. When the annotations of the second annotator were scored against the annotations of the first, an F-score of 82% was achieved. When credit was given for overlapping annotations, this rose to 96%.

Note that, although of sufficient size for evaluation purposes, ImmunoTome is much smaller than standard training corpora. It is therefore not large enough to facilitate the retraining of existing tagging tools.

2.2 Tagger evaluation

To calculate approximate upper and lower bounds on the performance of different taggers, we assessed their performance using both "strict" and "sloppy" matching criteria. When strict criteria are applied, a tagger is required to match a given protein name exactly to score a "hit". When sloppy criteria are applied, the tagger scores a "hit" provided part of the protein name is matched.

The extent to which exact matching is required in practice is application-dependent. However, in terms of the fair evaluation of tagger performance, the use of strict matching criteria has a clear disadvantage; the performance of a tool will vary significantly depending on essentially arbitrary choices made by the annotators of the evaluation corpus (e.g. is the word "mouse" part of the protein name in the phrase "mouse oxytocin"?). With sloppy matching criteria, on the other hand, there is a risk that a tool will gain credit even when it has missed the core part of a protein name (e.g. if it exclusively annotated either the word "activated" or "protein" in the phrase "activated ras-1 protein").

We investigated a random set of 100 tagged protein names that count as hits with sloppy criteria, but as misses with strict criteria. In every case the core of the protein name was contained within the annotation. In 18 cases an erroneous word had been incorporated (e.g. the word "namely" in "namely IFN-gamma"), in 21 cases the discrepancies were associated with the conjunction "and" (e.g. "CD4 and CD8 coreceptors" is annotated as a single name in ImmunoTome, but as two proteins by the tagger), and the remainder involved more-or-less legitimate extensions to, or contractions of, the name as annotated in ImmunoTome (e.g. "Ag antivenin" instead of "antivenin").

3 Results

3.1 Comparative performance of taggers

The performance of our chosen taggers on four corpora is given in table 1. These results show that ABNER is the best-performing tagger on all corpora, with the BioCreAtivE version of ABNER registering the best scores on ImmunoTome using both strict and sloppy matching criteria. All the other taggers show a significant drop in performance when evaluated on ImmunoTome.

	Y	J	P	I
Sloppy matching criteria				
ABNER (B)	80.3	76.0	85.3	78.3
ABNER (G)	73.9	84.1	80.1	69.5
NLProt	N/A	70.8	81.2	66.1
LingPipe	65.3	79.1	67.4	53.7
Gapscore	80.5	68.6	81.3	56.6
Strict matching criteria				
ABNER (B)	54.2	60.8	59.4	54.0
ABNER (G)	48.4	67.9	62.0	47.8
NLProt	N/A	45.8	59.7	43.8
LingPipe	43.4	62.8	47.0	33.6
Gapscore	57.4	38.3	52.9	30.7

Table 1. The F-scores produced by five taggers when applied to four corpora. Abbreviations are as follows: Y = Yapex; J = JNLPBA evaluation corpus; P = ProSpecTome (Kabiljo *et al*., 2007); I = ImmunoTome; B = BioCreAtivE; G = GENIA. As NLProt was trained on the Yapex corpus, no fair test score can be provided for this combination.

ImmunoTome differs from the other corpora in two important respects: it contains texts from a distinctive sub-domain; and it comprises full-text articles rather than abstracts. In order to shed light on the relative impact of these differences,

we independently evaluated the taggers that were not trained on GENIA using the subset of GENIA abstracts containing the annotations *CD4* or *CD8* (86 abstracts from a total of 2,000). The results are shown in table 2.

	Full GENIA corpus	Subset of GENIA
Sloppy matching criteria		
ABNER (B)	79.7	83.5
NLProt	74.2	77.1
Gapscore	73.8	77.0
Strict matching criteria		
ABNER (B)	64.3	66.8
NLProt	49.7	54.8
Gapscore	40.8	48.1

Table 2. The F-scores of three taggers – none of which were trained using the GENIA corpus – applied to GENIA and to an immunological subset of GENIA.

These results suggest that taggers find it easier to correctly identify protein names from the immunological sub-domain than from general biomedical texts. To explore possible reasons for this, we analyzed the most frequently-occurring protein names in three corpora (table 3). Note that the top ten protein names in ImmunoTome account for 43% of the total annotations in that corpus – much higher than the equivalent figures for Yapex and GENIA (6% and 9% respectively). This is to be expected given the repetitious use of protein names in full-text articles.

Yapex	GENIA	ImmunoTome
NF-kappa B (28)	NF-kappa B (862)	CD4 (518)
Tat (27)	NF-kappaB (542)	CD8 (348)
CD4 (26)	IL-2 (535)	TCR (156)
p53 (26)	transcription factors (332)	TRX1 (143)
NF-kappaB (23)	AP-1 (322)	TCR-αß (74)
GM-CSF (22)	IL-4 (314)	CD40 (59)
IL-2 (22)	transcription factor (283)	TNF (58)
SMN (22)	TNF-α (245)	IFN-γ (53)
IL-6 (22)	IFN- γ (227)	CD40L (52)
SUMO-1 (21)	cytokines (200)	2C TCR (51)

Table 3. The top ten occurring protein names in three corpora. The number of occurrences of each name is recorded in parentheses. Different forms of the same name (e.g. "NF-kappa B" and "NF-kappaB") are recorded separately.

From table 3 it appears that names of forms that are likely to prove comparatively easy for taggers to identify are more prevalent in ImmunoTome. In particular, names made from a combination of upper-case letters and digits account for six out of ten names on the ImmunoTome list, compared with four for Yapex and three for GENIA. On the other hand, names in lower case (easily confused with general vocabulary) or title case (easily confused with generic proper names) are more prevalent in Yapex and GENIA. (Note that the appearance of general references to proteins – e.g. "cytokines" – exclusively on the GENIA list is attributable to the annotation guidelines associated with that corpus.)

3.2 Information content of ImmunoTome

The distribution of protein names across the different sections of the full-text articles in ImmunoTome are summarized in table 4. Of the total number of protein names, less than 5% are found in the abstracts. When the same calculation is performed for distinct protein names, less than 10% are found in the abstracts. Unsurprisingly, the number of protein names uniquely found in the abstracts of ImmunoTome is very low (though, perhaps surprisingly, non-zero).

	TA	I	M	RD
total words	2262	6331	7418	33786
total annotated	171	484	397	2437
total distinct	84	212	275	518
total distinct & exclusive	14	111	189	359
annotated / words (%)	7.6	7.6	5.4	7.2
distinct / words (%)	3.7	3.4	3.7	1.5
exclusive / words (%)	0.6	1.8	2.5	1.1

Table 4. The information content of the ImmunoTome corpus broken down by section. Abbreviations are as follows: TA = Title + Abstract; I = Introduction; M = Materials and Methods; RD = Results + Discussion. The "distinct & exclusive" total records the number of distinct protein names that are exclusively found within a given section.

With respect to the density of information, the results are less clear-cut. Ultimately it makes sense to select the most relevant sections for a given application, and relevance is not something that can be assessed by a simple analysis of name density (Shah *et al*., 2003).

Whatever the application, it is certainly worth taking into account the variable performance of protein taggers on the different sections of full-text articles. This is summarized for ImmunoTome in table 5.

There are two notable features of these results. Firstly, all taggers except Gapscore perform best on the Introduction section, in spite of the fact that all the taggers except Gapscore were trained on abstracts. This is a surprising result and one that warrants further investigation.

	TA	I	M	RD
ABNER (B)	79.4	83.1	72.4	78.2
ABNER (G)	74.2	75.3	63.2	69.2
NLProt	67.4	75.9	48.0	67.4
LingPipe	60.4	61.1	52.1	52.1
Gapscore	63.2	54.6	46.4	46.4

Table 5. The F-scores of five taggers on different sections within the ImmunoTome corpus evaluated using sloppy matching criteria. Abbreviations are the same as for table 4.

Secondly, all tools perform worst on the Materials and Methods section. A common problem here is the relatively high density of proper names such as *Pharmingen* and *Sweden*.

4 Conclusion

That ABNER (BioCreAtivE) proves to be the best single tagger when applied to ImmunoTome reinforces the conclusion we reached elsewhere (Kabiljo *et al*., 2007). It may be significant that this version of ABNER did much better than the version trained on GENIA. Further investigation is needed to decide whether this is attributable to the content of the BioCreAtivE corpus, to its annotation guidelines, or to other factors.

This is, we believe, an important finding. The construction of new training corpora is very time consuming, hence it is highly unlikely that multiple training corpora focusing on specific biomedical sub-domains will become available in the foreseeable future. In these circumstances, researchers wishing to perform named entity recognition in a biomedical sub–domain have little option but to use one or more existing taggers. Our results show that, at least for the sub-domain of immunology, this does not lead to a large drop in performance – provided that the chosen tagger is ABNER (BioCreAtivE).

Our results also show that taggers have particular problems when annotating the Materials and Methods sections in ImmunoTome. This is likely to be true more generally, and suggests that for some applications it is sensible to exclude Materials and Methods sections altogether.

Finally, using multiple corpora to evaluate the performance of different protein taggers potentially gives us deeper insights into their relative performance. From this perspective, ImmunoTome complements existing corpora, and will offer a new dimension to future analyses. In this role, the usefulness of ImmunoTome is enhanced by the provision of explicit annotation guidelines, and the assessment of inter-annotator agreement reported above.

Acknowledgements

The research of R.K. is supported by European Commission under FP6-2004-IST-4 contract no. 028069 (the ImmunoGrid project) and the UK ORSAS scheme. We would like to thank Diana Stoycheva, University of Heidelberg, for her help in creating the ImmunoTome corpus.

References

Jeffrey T. Chang, Henrich Schutze and Russ B. Altman. 2004. GAPSCORE: finding gene and protein names one word at a time.*Bioinformatics*, 20(2):216-225.

Kristofer Franzén, Gunnar Eriksson, Fredrik Olsson, Lars Asker, Per Lidén and Joakim Cöster. 2002. Protein names and how to find them. *International Journal of Medical Informatics*, 67(1-3):49-61.

Renata Kabiljo, Diana Stoycheva and Adrian J. Shepherd. 2007. ProSpecTome: a new tagged corpus for protein named entity recognition. *Proceedings of the Annual Meeting of the ISMB BioLINK Special Interest Group on Text Data Mining*, Vienna, 19 July 2007, 24-27.

Jin-Dong Kim, Tomoko Ohta, Yuka Teteisi and Jun'-ichi Tsujii. 2003. GENIA corpus - a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(suppl. 1):i180-i182.

Jin-Dong Kim, Tomoko Ohta, Yoshimasa Tsuruoka, Yuka Teteisi and Nigel Collier. 2004. Introduction to the Bio-Entity Recognition Task at JNLPBA. *Proceedings of JNLPBA 2004*, 70-75.

Sven Mika and Burkhard Rost. 2004. Protein names precisely peeled off free text. *Nucleic Acids Research*, 32(Web server issue): W634-W637

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

Parantu K. Shah, Carolina Perez-Iratxeta, Peer Bork and Miguel A. Andrade. 2003. Information extraction from full text scientific articles: Where are the keywords? *BMC Bioinformatics*, 4:20.

Alexander Yeh, Alexander Morgan, Marc Colosimo and Lynette Hirschman. 2005. BioCreAtIvE Task 1A: gene mention finding evaluation. *BMC Bioinformatics,* 6(Suppl 1):S2.

Towards Knowledge Discovery through Automatic Inference with Text Mining in Biology and Medicine

Hee-Jin Lee and Jong C. Park
Computer Science Division, KAIST
373-1 Guseong-dong, Yuseong-gu, Daejeon 305-701 Republic of Korea
{heejin,park}@nlp.kaist.ac.kr

Abstract

Field experts in biology and medicine search the literature for state-of-the-art results and occasionally discover knowledge through manual inference on published causal relations. However, the results of such inference cannot be sufficiently accurate and/or complete, as the domain of published relations is rather huge. In this paper, we introduce an automatic inference system, BioDetective, which works on literature-mined qualitative causal information in biology and medicine. BioDetective provides proofs for such qualitative causal information, and predicts the existence of new causal information, if there is any. The system is tested with a case study, where literature-mined information about protein regulation is utilized to come up with new knowledge.

1 Introduction

Field experts in biology and medicine search the literature for state-of-the-art results and occasionally discover knowledge through manual inference on published causal relations. For example, a biomedical scientist who seeks new treatments for a disease may search the literature for information about biological or medical entities already known to be related to the particular disease, as well as about causal relations that are known to involve these entities. By inferring over the combined effect of such causal relations, she may be able to discover novel (and possibly indirect) causal relations between the disease and some molecules and/or biological conditions. She may also use such novel causal information towards finding effective drugs for the disease. Such an approach to knowledge discovery through manual inference on literature-mined information would be a good way to reduce the number of repeated experiments that are based purely on intuition, which may turn out to be not only time-consuming but also literally quite expensive.

However, the fraction of information that can be manually examined this way is much limited. For one, the experts may not be able to locate the connecting information that would have been easily identified if the available body of knowledge were larger. Even when a larger body of knowledge is taken into account, manual inference is susceptible to mistakes due to the complexity of the involved inference. Automated inference on a dataset of literature-mined information will certainly help the field experts to cover more information with fewer mistakes.

In this paper, we introduce an automatic inference system, BioDetective, for literature-mined qualitative causal information in biology and medicine. Given a collection of causal information and other related literature-mined information as the input dataset, BioDetective can check if the input dataset supports a new, hypothetical causal relation between known biological (or medical) entities. We believe that this helps field experts to discover new knowledge from literature-mined information. In order to assess the performance, we tested the system with a case study, where literature-mined information about protein regulation is employed.

We review other inference systems in Section 2, introduce BioDetective in Section 3, describe our case study in Section 4, and show concluding remarks in Section 5.

2 Related Work

Notation	Types			Description	Flow of effects
	x	y	z		
z			SE	Biological Process	Ⓩ
z			SEM	Molecule	Ⓩ
z			SE	External control or disease	Ⓩ
z → y		M	SEM	Modification to molecule	Ⓩ ← Ⓨ
x ← z → y	M	M	SEM	Binding	Ⓧ → Ⓩ ← Ⓨ
x — z → y	S	E	E	Inhibition	Ⓧ → Ⓩ → Ⓨ
x — z ▷ y	S	E	E	Induction	Ⓧ → Ⓩ → Ⓨ
x — z ‖▷ y	S	E		Necessary condition	Ⓧ → Ⓩ → Ⓨ
x — z → ø	S	S	E	Degradation	Ⓧ ← Ⓩ

Table 1. Symbols used in DPL. S is for states, E for events, and M for molecules.

BioDetective handles high level concepts together with molecular level concepts, as the information in the literature is often at a level higher than the molecular level. The system can also accommodate new kinds of concepts, unknown to the system beforehand. These two characteristics, unavailable from current systems, of which some are reviewed below, facilitate the system to produce new information from literature-mined information, by enabling the system to connect information which would be considered otherwise unrelated.

BioSigNet-RR is a system for representing and reasoning about signaling networks (Baral et al., 2004), and can deal with four kinds of queries on the cellular signaling network, but does not seem to be easily adaptable to other sub-domains of biology. BIOCHAM is a software tool for modeling biochemical systems (Calzone et al., 2006), and can conduct analysis and simulation of biological models, but requires data to be only at the molecular level. Pathway Logic is an approach to modeling biological entities/processes based on rewriting logic (Eker et al., 2005), for the analysis of models of signal transduction networks, but does not appear applicable to literature-mined information with higher level concepts.

3 BioDetective

Given a database of biological causal information and a query describing a causal relation, BioDetective checks if the input dataset supports the causal relation[1]. The structure of BioDetective is shown schematically in Figure 1.

The input database should contain a dataset that forms a causal network, to be defined in Section 3.1. The information in the input database is used by the model generator to generate a model

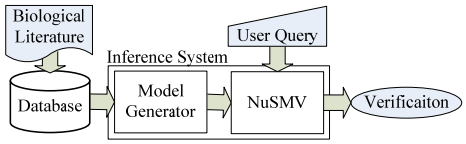


Figure 1. Structure of BioDetective

of a concurrent system, which follows rules in Section 3.2. The generated model is provided to NuSMV (Cimatti et al., 2002), an open source model checker. A causal relation stated as a temporal query as explained in Section 3.3 is provided to NuSMV for verification.

3.1 Causal network as input

To describe the datasets that can be used as input to BioDetective, we define Diagrammatic Pathway Language (DPL) (cf. Park and Park, 2005).

DPL is a set of pathways, where a pathway is defined as a set of connected symbols, which are any of the symbols of 9 types shown in Table 1, where positions marked by x and y are instantiated by other connected symbols. DPL follows the style as proposed by Kohn and others (2006).

A collection of biological or medical information forms a causal network when each piece of such information is represented with a corresponding symbol in DPL. We assume that the body of information comprising the input dataset forms a causal network.

Notice that the information in a causal network includes higher level concepts such as causal relations, and non-causal concepts such as diseases and biological processes.

3.2 Rules for concurrent systems

The concurrent system of a causal network generated by the model generator consists of biological or medical entities represented each with a symbol in the pathway of the causal network. Each entity of the concurrent system can either be present or absent. The status of entities may change simultaneously over time according to

[1] The system is based on the abstract description of a qualitative formalization framework by Park and Park (2005), implemented here with extensions for automated execution.

Rule name	Description
Environment Assumption	1. When a non-external molecule A is not the target of any induction or inhibition, the molecule is considered initially present. 2. When a disease or external control or a molecule is the target of any induction or inhibition, it is considered initially not present. 3. Otherwise, the status of the biological entity is considered not initially determined.
Implicit Necessary Condition	The presence of participants of a biological entity is a necessary condition for the biological entity.
Dynamic Inference	- Biological entity X will be present if 1. for some A that induces X, A is present, and 2. for all B that inhibits X, B is absent, and 3. for all necessary conditions C for X, C is present. - Biological entity X will be absent, otherwise.
Inertia	Once a biological entity becomes present by the Dynamic Inference rule, it remains present unless it is interfered.

Table 2. Rules for concurrent systems.

the causal relations involving the entities. Rules for such status changes are summarized in Table 2. The status changes of all the entities in a concurrent system reflect the combined effect of all the causal relations in the causal network.

Note that the rules are not specific to the kind of entities involved in the chain of causal relations, and that the inference system can easily accommodate new types of biological entities.[2]

3.3 Causal relations as queries for NuSMV

We use NuSMV to compute the temporal properties of a concurrent system to verify causal relations of interest. A causal relation between two biological or medical entities is stated as temporal properties in Linear Temporal Logic (LTL), using two LTL operators 'in the future (*F*)' and 'globally (*G*)'. Given a formula in LTL as a query, NuSMV returns *true* if the concurrent system of the input causal network has the queried temporal property, which means that the queried causal relation is supported by the input dataset. Inducing and inhibiting relations between entities A and X are stated respectively as follows.

- Induction of *X* by *A*: *A -> F G X*
- Inhibition of *X* by *A*: *A -> F G !X*

If the causal relation is not shown explicitly with a symbol in the pathway of the causal network, the relation is considered indirect. An indirect causal relation verified by BioDetective would work as a novel piece of information, obtained by connecting pieces of known information.

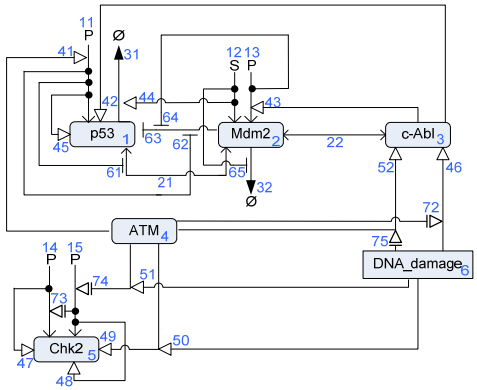


Figure 2. The causal network constructed as in Section 4. Every entity is marked with its ID number.

4 Using BioDetective: A case study

In this section, we demonstrate possible uses of BioDetective with a case study, where the system takes literature-mined information as input and produces new knowledge, if there is any.

Construction of a causal network: We constructed a causal network semi-automatically so that it can be used as input for our case study, as shown in Figure 2. Using BioIE (Kim and Park, 2004), an information extraction system specialized for biology and medicine, we extracted 109 descriptions of interactions and causal relations between ATM, Mdm2, Chk2, c-Abl, p300 and p53, from MEDLINE. We then manually examined the extracted pieces of information to construct a causal network and manually stored the network in an SQLite database.[3]

Phase 1 – Resolving multiple representations: There are cases where a causal relation is represented explicitly in a causal network, but is also represented by paths of other causal relations and interactions in the same network. These cases possibly result from the mixed nature of

[2] We believe that the 9 types of biological entities currently handled by the system form a complete set of types, but new types such as 'phenotypes' may still be introduced if needed. Existing types such as 'induction' may also be split into lower level types, such as 'induction by transcription'. However, the rules themselves remain unchanged.

[3] The causal network contains 34 biological entities; We did not use all the extracted information.

natural language descriptions, each describing the same event with a different level of detail.

We utilized BioDetective to detect these cases. Given causal relation I, where A is a direct cause for B to change, we collected all the causal relations and interactions that transfer the status of A to B, to construct a subnetwork. This is done by using the information in the last column of Table 1. We then used BioDetective to see if the subnetwork supports I. If BioDetective returns *true*, the subnetwork is interpreted as representing the same event as I, but at a level more detailed than the one suitable for representing I.

We applied the procedure above to the input causal network as shown in Figure 3. We found five explicit causal relations having a subnetwork of the same effect. One of them is the inducing relation 42, 42 being the ID number in Figure 3, where the corresponding subnetwork consists of biological entities 1, 63, 64, 13, 43, 2, and 3. This multiplicity was evidenced by the following sentence found manually from the literature.

- Phosphorylation of Mdm2 by c-Abl impairs the inhibition of p53 by Mdm2, hence defining a novel mechanism by which c-Abl activates p53. [PMID: 12110584]

We removed all the five explicit causal relations to use the modified network in phase 2.

Phase 2 – Finding sufficient conditions for events to happen: If we consider a causal network of literature-mined information as a qualitative model of a biological system, and use a closed world assumption, we can find a sufficient condition for a biological or medical event to happen, using BioDetective.

The sufficient condition for a biological or medical entity X to be present can be found by searching for the initial configuration A of the input causal network that supports the query 'A -> F G X'. For this purpose, the causal network in Figure 3, cleaned up at the first phase of the case study, was used. One of the sufficient conditions found by the system is shown below.

- Absence of Mdm2 at the initial time is a sufficient condition for p53 to be present.

We plan to improve the system performance further by selecting a subset of the configurations of the initial status of the input network.

5 Concluding Remarks

We introduced BioDetective, an automatic inference system that deals with qualitative causal information in biology and medicine. The system is suitable for producing new information by meaningfully connecting existing pieces of information in the literature.

The system is utilized in a case study, where literature-mined information is collected and processed to obtain new knowledge. The case study showed the possibility that the system is applicable for various tasks for integration and utilization of the literature-mined information.

Acknowledgments

We thank Robert Kueffner and Hodong Lee for valuable comments on earlier versions of this paper. This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD) (KRF-2007-313-D00738) and by the research funding on an opportunistic project by Microsoft Research Asia.

References

Chitta Baral et al. 2004. A knowledge based approach for representing and reasoning about signaling networks, *Bioinformatics*, 20:i15-i22.

Laurence Calzone et al. 2006. BIOCHAM: an environment for modeling biological systems and formalizing experimental knowledge, *Bioinformatics*, 22(14):1805-1807.

Alessandro Cimatti et al. 2002. NuSMV 2: An open-source tool for symbolic model checking, *In Proceedings of CAV 2002*, 27-31.

Steven Eker et al. 2005. Pathway Logic: executable models of biological networks, *Electronic Notes in Theoretical Computer Science*, 71: 144-161.

Jung-jae Kim and Jong C. Park. 2004. BioIE: retargettable information extraction and ontological annotation of biological pathways from the literature, *Journal of Bioinformatics and Computational Biology (JBCB)*, 2(3):551-568.

Kurt W. Kohn et al. 2006. Molecular interaction maps of bioregulatory networks: A general rubric for systems biology, *Molecular Biology of the Cell*, 17:1-13.

Il Park and Jong C. Park. 2005. Modeling causality in biological pathways for logical identification of drug targets, *The 2005 International Joint Conference of InCoB, AASBi and KSBI (Bioinfo 2005)*, 373-378.

Why Biomedical Relation Extraction Results are Incomparable and What to do about it

Sampo Pyysalo,[1] **Rune Sætre,**[2] **Jun'ichi Tsujii**[2] and **Tapio Salakoski**[1]
[1]Turku Centre for Computer Science (TUCS) and Dept. of IT, University of Turku
20014, Turku, Finland
{sampo.pyysalo,tapio.salakoski}@it.utu.fi
[2]Department of Computer Science, the University of Tokyo, Japan
Bunkyo-ku, Hongo 7-3-1, Tokyo, Japan
{rune.saetre,tsujii}@is.s.u-tokyo.ac.jp

Abstract

A large number of biomedical relation extraction methods, targeting for example protein-protein interactions (PPI), have been introduced in the preceding decade. However, the performance figures reported for these methods vary enormously, and results are largely incomparable across different studies. In this paper we study reasons leading to this situation and propose a solution to resolving them.

1 Introduction

Evaluation results for biomedical relation extraction methods vary greatly and are largely incomparable across different studies. This makes it difficult to assess what are the best tools, methods, techniques and general approaches to the task. A number of recent studies have brought to light several issues leading to this incomparability. In this paper we collect together these findings and discuss several other aspects of relation extraction experiments that may introduce unwanted variance into evaluation results. After reviewing the problems, we propose a solution to the known issues.

We assume throughout the paper the common task setting where relations are to be extracted by identifying entity pairs for which the relation holds, e.g. two proteins that are stated to interact. While a machine-learning perspective is involved in some parts of the discussion, most of the problems can occur for any extraction approach. We assume that evaluation aims to be able to establish differences in the performance of compared methods on the order of a few percentage units or less, a level of accuracy at least implicitly assumed in many comparisons of domain extraction methods but, as we shall discuss next, far from systematically achieved at present.

2 The problems

2.1 Different corpora

In a recent study of biomedical relation extraction performance across five corpora, Pyysalo et al. (2008) demonstrated that evaluation results for a single method on different corpora may vary up to 30%, and found a 19% average performance difference on the corpora. These differences stem in part from different definitions of what should or should not be extracted as a protein-protein interaction, which leads to differing positive/negative distributions of candidate relations: for example, the LLL corpus (Nédellec, 2005) contains 164 "true" (positive) relations out of 330 possible entity pairs, giving an "all-true" baseline performance of 66% F-score[1], while for the AIMed corpus (Bunescu et al., 2005) these figures are approx. 1000 positive out of 5800 candidate pairs for a baseline performance of 29% F-score.

While differing extraction targets are, in general, a benefit for evaluation—extraction approaches should be able to learn different targets—these differences render (unqualified) evaluation results from different corpora incomparable. Below, we will only consider factors complicating evaluation on a shared corpus.

[1]Assigning all candidates into the positive class gives a r(ecall) of 100% and a p(recision) of $\frac{c_p}{c_p+c_n}$, where c_p and c_n are the number of positive and negative candidates (resp.); F-score is $\frac{2pr}{p+r}$.

2.2 Corpus processing

Biomedical corpus annotation is rarely, if ever, distributed in a form that would explicitly specify the set of candidate relations. Instead, candidates must be generated, often from annotation that only specifies entities and positive relations. Negative relations are typically generated under the closed-world assumption. Along with various other details of annotation schemes, this opens the door to varying interpretations of single corpora.

2.2.1 Number of generated examples

With complex annotations including for example nested or noncontinuous entities, corpus annotation can allow for strikingly different numbers of positive and negative relations: Sætre et al. (2008) note that the AIMed corpus has been variously interpreted as containing between 951 and 1071 positive relations with 4026–5631 negative ones. For the most favorable combination (1071 positive, 4026 negative) the all-true baseline would stand at 35% and for the least favorable (951/5631) at 25% F-score. Thus, different preprocessings of the corpus can give a very large absolute difference even for a trivial baseline, rendering results for different preprocessings of the corpus incomparable.

A particular difficulty is presented by the existence of self-interactions, where an annotated (positive) relation involves only a single entity. While the AIMed corpus contains 54 such interactions, most studies on AIMed simply ignore their existence, since generating candidate relations involving only single entities would increase the number of negative candidates by thousands and lead to a considerably more difficult positive-negative ratio. A similar situation occurs when extracting directed relations: if each pair of entities is used to generate two directed candidate relations, the number of negative examples will more than double.

2.2.2 Entity name blinding

Biomedical corpora often focus on limited subdomains, either by design or due to bias introduced from document selection procedure (e.g. documents cited as evidence in an interaction database). Consequently, corpora can contain a disproportionate amount of relations between particular entities, which can be "memorized" by a learner if it is allowed to see their names. For example, in an experiment on the AIMed corpus we got an F-score of 33% when *only* the names of the candidates were used as features. As the all-true baseline is 30% for our version of the corpus, this suggests that memorizing names can provide a small but non-negligible benefit, again leading to diverging results. Extraction methods should be able to detect relations between entities whose names have not occurred in their training data—indeed, such novel interactions are more interesting than those already annotated. Thus, performance increments based on knowing the names of the entities involved do not reflect real benefits of extraction methods.

A related issue arises on corpora involving nested entities. For example on the AIMed corpus, the dataset applied in (Giuliano et al., 2006) appears to have been preprocessed so that nested entity names were treated differently depending on whether the inside entity was part of a true relation or not. For example, in the sentence *Cloning and functional analysis of [$_1$BAG-1] : a novel [$_2$[$_3$Bcl-2]-binding protein] with anti-cell death activity* there are three potential pairs (1,2), (1,3) and (2,3), but in the Giuliano dataset only two pairs for this sentence are given, one false pair, (1,3), and one true pair, (1,2), where the representation of the latter does not involve marking the tokens *-binding protein* as belonging to a protein name (and thus blinding). The negative candidate pair (2,3) is excluded in this case. Removing negative nested protein names raises evaluated performance in terms of F-score by increasing the positive/negative ratio. However, this way of preprocessing the data should not be performed unless there is a way to know in advance whether a nested entity is involved in a relation or not before running the extraction method. Comparison of evaluations where one employs such information and the other does not may not yield meaningfully comparable results: Airola et al. (2008) ran the method published by Giuliano et al. (2006) on a differently blinded version of AIMed and reported a 52.4% F-score, over 6% points lower than the 59.0% reported by Giuliano et al.

2.3 Experimental setup

There are numerous potential pitfalls in setting up a relation extraction experiment, in particular when it involves machine learning. Two frequently encountered issues relate to the role of training and test sets in evaluation.

2.3.1 Isolating training and test data

To establish a meaningful estimate of generalization performance, the training and test sets must represent independent samples: test data that resembles the training data more than the overall distribution benefits overfit learners and leads to overestimation of performance.

Sætre et al. (2008) observed that a number of biomedical relation extraction studies performed cross-validation by first preprocessing the data to form all the possible candidate pairs of related entities, which were then randomly split into different sets for training and evaluation. In this procedure, pairs from the same sentence ended up being used both for training and testing within a single fold. Since the features from two neighboring pairs in a sentence are practically identical, this was shown to lead to an 18% points overestimation of the F-score performance compared to a more realistic setting. In the realistic test setting, all the data from a single abstract is kept together through the whole processing pipeline, to avoid using it both for training and testing in the same fold.

2.3.2 Parameter selection

The data on which methods are tested should, ideally, represent completely new, unseen data. While this ideal is rarely achieved, a small number of tests on the whole dataset is unlikely to cause much bias. However, experiments are often set up to include repeated, systematic tests on the entire dataset, of which the best result is reported. Perhaps the most frequent such setup arises from parameter selection, e.g. using cross-validation on the entire corpus. Especially when the parameter space is multi-dimensional and the data set is small, this approach can find considerable benefit from identifying "spikes" in the parameter space. Evaluation necessarily involves some random variation for different parameter settings, and a parameter selection protocol that allows the test set to be seen will yield an overestimate of performance relative to the magnitude of that variation. On smaller corpora (e.g. LLL), random effects changing the assignment of just a few examples can already make a percentage unit difference in results.

A related issue arises from picking the best point (e.g. in terms of F-score) from a precision-recall curve generated for a single extraction method with fixed overt parameters. This corresponds to implicitly optimizing a classification threshold parameter, again with reference to the whole dataset. When comparing methods with otherwise similar performance, these differences can cause misleading results: Using the method of Airola et al. (2008) on AIMed, picking the optimum threshold was estimated to provide at least a 2% overestimate over the more realistic setting of selecting the threshold on the training data.[2]

2.4 Metrics

Even when the same corpus, preprocessing, experimental setup, and metric are applied, differences arising from the details of how the metric is calculated can cause results to deviate.

2.4.1 Extracting Identical Relations

A relation is typically taken to be correctly extracted if the (unordered) pair of related entities is identified. However, this definition leaves open a question relating to entity identity: are two mentions of the same name one or two entities, and consequently, should two relations annotated between the same two names both be extracted, or does it suffice to find either one?

Giuliano et al. (2006) termed two answers to these questions One Answer per Occurrence in a Document (OAOD) and One Answer per Relation in a Document (OARD): here the OAOD criterion requires each mention to be extracted, while OARD only demands that each unique pair of names is identified. They found that an otherwise identical evaluation yielded an F-score of 59% under the OAOD criterion and 64% under OARD, indicating that results evaluated using different criteria cannot be directly compared.

The two alternatives studied by Giuliano et al. are not the only ones possible: we might propose One Answer per Sentence, One Answer per Corpus, One Answer per (cross-validation) Fold, or One Answer per Journal. While one might argue that extracting each relation from the corpus once suffices for some practical applications, we take the view that from the evaluation perspective the specific names (between which relations are stated) are of secondary importance and suggest that each relation be considered. That is, One Answer per Occurrence; from this perspective, the "D" in "OAOD" is superfluous.

[2] Thanks to Antti Airola for running this number for us.

2.4.2 Averages

How averages are calculated is a lesser, but not negligible, issue. This question often arises from cross-validation, where two basic alternatives are available: either calculate performance for each fold separately and average the results (macroaveraging), or pool the answers and calculate one result for the entire dataset (microaveraging). Different choices might cause non-trivial differences in otherwise identical setups for small corpora: for example, when examples are carefully divided into cross-validation folds on the document level, some test sets can contain documents with unusually high numbers of entities and thus of candidate relations. With macroaveraging, folds with a large number of relations will contribute equally to the final result as folds with fewer, whereas if results are pooled the contributions of folds will be inequal, but each relation will contribute equally. As the number of candidate relations grows quadratically with the number of entities in a given context and the growth of positive relations is likely to be slower, we would expect folds with more relations to represent more difficult problems in terms of metrics sensitive to the positive/negative distribution (e.g. F-score) and thus macroaveraged results to be higher.

3 A proposal for a solution

The problems discussed above highlight a need for standardization to establish meaningful comparisons between different relation extraction method evaluations. Before these issues are addressed to some extent, the only direct comparisons between methods that can be meaningfully performed are those done within a single study (or at least by the same authors) and those from shared tasks. The incomparability comes at a great cost to the community, as reimplementation is often the only way to reliably determine the relative merits of proposed methods.

We do not expect that specific choices to the many alternatives discussed could be enforced by fiat. Instead, we propose a positive solution: we have constructed a standard dataset containing data derived from different corpora, building on the unification of five corpora under a common format by Pyysalo et al. (2008). We have extended this work by including explicit candidate pairs with blinded protein names, thus addressing the issues in corpus processing. Further, predefined train/test splits are provided, and the distribution of the dataset is accompanied with evaluation scripts that implement the basic metrics in a standardized way, thus eliminating possible differences arising from metric application. The data and software is freely available from `http://mars.cs.utu.fi/PPICorpora`.

4 Conclusion

We have discussed a number of issues in biomedical relation extraction system evaluation that complicate, or even prevent, meaningful comparison of reported results, and we proposed a solution to address these issues. We believe that the proposed dataset and evaluation approach can serve as a step toward stable, reliable evaluation of biomedical relation extraction methods.

Acknowledgments

The work was partially supported by the Academy of Finland, Grant-in-Aid for Specially Promoted Research (MEXT, Japan) and Genome Network Project (MEXT, Japan).

References

Antti Airola, Sampo Pyysalo, Jari Björne, , Tapio Pahikkala, Filip Ginter, and Tapio Salakoski. 2008. A graph kernel for protein-protein interaction extraction. In *Proceedings of the BioNLP'08*, pages 1–9.

Razvan C. Bunescu, Ruifang Ge, Rohit J. Kate, Edward M. Marcotte, Raymond J. Mooney, Arun Kumar Ramani, and Yuk Wah Wong. 2005. Comparative experiments on learning information extractors for proteins and their interactions. *Artif Intell Med*, 33(2):139–155.

Claudio Giuliano, Alberto Lavelli, and Lorenza Romano. 2006. Exploiting shallow linguistic information for relation extraction from biomedical literature. In *Proceedings of EACL'06*.

Claire Nédellec. 2005. Learning language in logic - genic interaction extraction challenge. In *Proceedings of LLL'05*.

Sampo Pyysalo, Antti Airola, Juho Heimonen, Jari Björne, Filip Ginter, and Tapio Salakoski. 2008. Comparative analysis of five protein-protein interaction corpora. *BMC Bioinformatics*, 9(Suppl 3):S6.

Rune Sætre, Kenji Sagae, and Jun'ichi Tsujii. 2008. Syntactic features for protein-protein interaction extraction. In *Proceedings of LBM'07*, volume 319, pages 6.1–6.14.

Assessment of Modifying versus Non-modifying Protein Interactions

Dietrich Rebholz-Schuhmann[1], Antonio Jimeno[1], Miguel Arregui[1] and Harald Kirsch[1]
[1]European Bioinformatics Institute, Wellcome Trust Genome Campus, Hinxton, Cambridge, CB10 1SD, U.K.

Abstract

Motivation: The identification of events such as protein-protein interactions (PPIs) from the scientific literature is a complex task. One of the reasons is that there is no formal definition for the syntactical-semantic representation of the relations with which authors of manuscripts have to comply. In this study, we assess the distribution of verbs denoting binary relations between proteins using different corpora (AIMed, BioInfer, BioCreAtIve II) for protein-protein interactions and measure their performance for the identification of PPI events (in the BioCreAtIve II corpus) based on syntactical patterns. We distinguish modifying interactions (MIs) such as post-translational modifications (PTMs) from non-modifying interactions. We found that MIs are less frequent in the corpus but can be extracted at the same precision levels as PPIs.
Programmatic access to the text processing modules is available online (www.ebi.ac.uk/webservices/whatizit/info.jsf, http://www.ebi.ac.uk/Rebholz-srv/pcorral/).

1 Introduction

Since the innovative approach of (Blaschke et al., 1999), a number of solutions for the identification of binary relations such as protein-protein interactions (PPIs) have been proposed. Until today, no solution is yet publicly available that at the same time identifies from the scientific literature the protein and gene names (PGNs), links them to the concept id (CID) in the biomedical data resources (e.g., to the accession number in UniProtKb) and reads out the relation between two PGNs at a high precision rate (precision = # correctly identified results / all identified results). Several solutions have been proposed (see related work), including the one that is best-known and called iHOP (Hoffmann et al., 2005), but none of them offers a comprehensive approach.

In this research work we explore on the use of language in the scientific literature, in particular in annotated corpora for protein-protein interactions to better understand the use of verbs in this context. We follow the hypothesis that language representations for PPIs fall into different categories: (a) interactions with chemical modifications to one interaction partner ("modifying interaction", MI), and (b) interactions without such changes ("non-modifying interactions", NMI). The distinction between these types is motivated by the assumption that strong experimental proof for the MIs leads to explicit statements in the scientific literature reporting on the interaction (e.g., explicit mention of the interaction partners) and thus information extraction techniques will achieve better performances.

The evidence for the modifying interactions is any reporting of chemical changes linked to the interaction partners of the PPI. For example, methylation and demethylation and similarly phosphorylation and dephosphorylation as well as other types of chemical changes (e.g., acetylation, biotinylation) have to be considered here (see table 1). These modifications can be subsumed as posttranslational modifications (PTMs), which are a subcategory of PPIs. (Saric et al., 2006) have integrated these types of interactions into their work. Since the experimental evidence for the reporting of an interaction is linked to chemical changes which require modifying contact between the two proteins, it can be expected that the reported results is a proven protein-protein interaction.

The second group of reported protein-protein interactions forms the largest set and has been commonly used for the identification of PPIs (Temkin et al., 2003; Friedman et al., 2001; Blaschke et al., 1999). This group contains all reported results, where for example one protein activates or binds another protein. This set of interactions is relevant to molecular biologists searching for clues to reconstruct regulatory and signaling pathways in the cell.

The proposed categorization meets the demands from members of curation teams at the EBI that

require integration of different interaction types (modifying and non-modifying interactions) into public services (Protein Corral, unpublished). These services will now be properly assessed, after an appropriate evaluation corpus has been made available: the evaluation corpus for protein-protein interactions as part of the BioCreAtIve II challenge (Krallinger et al., 2007).

2 Methods

The identification of protein-protein interactions from the literature is a complex task, which is composed of named entity recognition for proteins, protein name normalization (i.e. identification of the correct CID) and the extraction of the relation between both entities. For the assessment we relied on the BioCreAtIve II corpus for the IPS task (347,749 sentences from 740 full-text documents), on the AIMed corpus (1,942 sentences from 255 abstracts) and on BioInfer (1,100 sentences from full-text) (Krallinger et al., 2007; Bunescu et al., 2005; Pyysalo et al., 2007). Only the BioCreAtIve corpus delivers a set of CID pairs for every contained document where the CID pair represents a protein-protein interaction.

2.1 Named entity recognition for proteins/genes

The identification of PGNs has been studied extensively (Morgan et al., 2007; Hakenberg et al., 2005; Hirschman et al., 2005). The identification of gene mentions has been solved to a precision close to 90% whereas the gene normalization is still ongoing work. In this work, the applied tagger (SPtagger) delivers CIDs as part of the NER task and is part of several TM solutions at the EBI (EbiMed, PCorral, MedEvi; Rebholz-Schuhmann et al., 2007a). It incorporates all protein names from UniProtKb/SwissProt and named entity recognition is mainly done by dictionary lookup under consideration of morphological variability, acronym resolution and basic disambituation (Tsuruoka et al., 2007; Gaudan et al., 2005; for SOAP Web services access see Rebholz-Schuhmann et al., 2007b).

2.2 Identification of protein-protein interactions

The identification of protein-protein interactions from the text is based on the modules of the Whatizit infrastructure (Rebholz-Schuhmann et al., 2007b) and through Protein Corral. Public access is granted to all modules that are used in this study. Most modules are implemented as Finite state automata (Kirsch et al., 2006). The basic NLP modules of the infrastructure comprise the sentenciser and a part-of-speech (PoS) tagger. The PoS tagger was trained on the British national corpus, but contains lexicon extensions for the biomedical concepts. Noun phrases (NPs) are identified with syntax patterns equivalent to "**DET (ADJ|ADV) N+**".

For our study we assessed tri-cooccurrence (3-CO) against syntactical patterns denoting a protein-protein interaction (SynP). 3-CO is performed on the stretch of a sentence. Any triplet of two proteins in combination with a verb mention in the following combinations is accepted: (1) "**PGN VP PGN**", (2) "**nomVP PGN PGN**", and (3) "**PGN PGN nomVP**", where nomVP is a nominalization of a verb phrase.

The module that identifies and highlights protein-protein interactions searches for phrases that contain a verb or a nominal form describing an interaction like bind or dimerization. The first set comprises all verbal expressions that report on chemical modifications of a protein: *acetylate, acylate, amidate, brominate, biotinylate, carboxylate, cysteinylate, farnesylate, formylate, "hydrox[iy]late", methylate, demethylate, "myristo?ylate", "palmito?ylate", phosphorylate, dephosphorylate, pyruvate, nitrosylate, sumoylate, "ubiquitin(yl)?ate"*. The second set of verbs consists of forms that report on interaction and regulation events: *associate, dissociate, assemble, attach, bind, complex, contact, couple, "(multi|di)meri[zs]e", link, interact, precipitate, regulate, inhibit, activate, "down[-]regulate", express, suppress, "up[-]regulate", block, contain, inactivate, induce, modify, overexpress, promote, stimulate, substitute, catalyze, cleave, conjugate, disassemble, discharge, mediate, modulate, repress, transactivate*. "Associate" does not denote any specific binding or transformation event.

The identification of noun phrases (NP) selects nouns in combination with adjective modifiers, including coordination of ADJ elements in front of a sequence of nouns. PGNs are treated as nouns. NPs do not include determiners (e.g., "novel orphan receptor TAK1"). Finally the protein-protein interaction patterns (PPI) are identified. They are basi-

cally combinations of the previously identified information, such as ***NP_P VP det? NP_P*** and ***NP_P VP det? NP of NP_P***, where NP_P is an NP that contains the identified protein and VP denotes verbal phrases including modal verbs. These construction rules for syntactical patterns lead to the selection of structures that are similar to tri-cooccurrence representations but generate higher precision. Similar structures have been proposed by (Huang et al., 2004). Nominalizations increase the recall for the identification of PPIs and follow the representation ***VP_NP "(of | with | between | through | from)" det? NP_P "(and | with | within | via | through | by)" det? NP_P***, where VP_NP is the nominalization of the verb form.

3 RESULTS

In the first step we analyzed all three available corpora, i.e. AIMed, BioInfer and BioCreAtIve, and extracted all verbs that co-occur with two mentions of a PGN. This resulted to the identification of 967 verbs for the BioCreAtIve corpus, 165 for AIMed and 162 for BioInfer. 90 were shared in all three corpora. Modal verbs (e.g., do, have) were only considered if they did not appear in combination with other verb forms. Apart from the domain-specific verbs (see method sections), a large list of general English verbs were extracted: encode, suggest, use, show, test. They are part of idiomatic phrases such as "we have shown that" or the "encoded protein". The first type is covered by our syntactical patterns if used as part of the textual protein interaction description.

From the list of NMI verbs 5 were not contained in AIMed (attach, catalyze, disassemble, modify, overexpress), 5 not in BioInfer (dimerize, down[-]?regulate, repress, substitute, transactivate) and 3 only in BioCreAtIve (conjugate, multimerize, up[-]?regulate). This shows that the BioCreAtIve corpus already by the number of provided sentences has the biggest coverage. It is a small surprise that "up-regulate" is not more commonly used.

Regarding the verbs categorized as MI only "phosphorylate" appeared in all three corpora and "acylate" in two corpora (i.e. not in AIMed). 4 verbs appeared only in the BioCreAtIve corpus (biotinylate, dephosphorylate, methylate, pyruvate). This leads to the result that MIs are preferrably reported in the full text document and at a low frequency. A complete Medline analysis has lead to the result that only a few verbs for MIs (biotinylate, dophosphorylate, hydroxylate, methylate, phosphorylate, pyruvate) are applied in conjunction with mentions of PGNs, whereas all verbs for NMIs are in use.

The following analysis focuses on the BioCreAtIve corpus only, since it is the largest corpus and the previous figures demonstrate that it provides the largest coverage of relevant verbs.

3.1 Comparison of NER tagging results

In our assessment, we considered the result of the protein-tagger as correct, if the right concept id (CIDs) was contained in the list of attributed CIDs. The resulting number is similar to the frequency of the identified named entities in the text and enables better comparison of results between the different methods (3-CO vs. SynP).

Table 2. (Processing full-text documents, One-CID) The table shows the results for the identification of CID pairs from the BioCreAtIve full text corpus for 3-CO and SynP.
SP (SwissProt-tagger), cs (case-sensitive), ci (case-insensitive), 3-CO (tri-cooccurrence), SynP (syntactical language patterns for PPIs)

	Predictions	Correct predictions	Precision	Recall	F-measure
SP-cs, 3-CO	12,771	408	3.2%	19.3%	5.5%
SP-cs, SynP	1,539	211	13.7%	10.0%	11.6%
SP-ci, 3-CO	15,823	609	3.8%	**28.8%**	6.8%
Sp-ci, SynP	2,078	358	**17.2%**	17.0%	**17.1%**

The evidence extracted with SynP is a true subset of the evidence from the 3-CO method leading to the result that about 50% (49.9%-58.8%) of the evidence from 3-CO can be confirmed by the approach using syntactical language patterns. This can be explained by the fact that the predictions are counts of unique CID pairs, which again can be represented by a number of instances in the document. The redundancy in the document counterbalances lower recall of the SynP methods over the 3-CO methods. In the next step we investigated into the distribution of the verb forms that were part of our two approaches.

According to our categorization, we find the following numbers for events representing MIs and NMIs (see table 3). The most correct predictions are reported in the set of NMIs (325) and the smallest number in the set of MIs (23). Altogether, MIs have a small contribution to all protein-protein interactions in the BioCreAtIve II corpus. The preci-

sion is for both types of events in the same range (18.5% and 17.2%, respectively). Similar results are gained when only processing the abstracts (MI: 7 agreements for 18 predictions; NMI: 64 agreements for 241 predictions).

To our surprise, the association of proteins has a significant contribution to the correct identification of relations between proteins. This result is unexpected, since the association of two proteins does not give any clues on the underlying relatedness of the proteins, i.e. a relation based on binding, regulatory or transformational effects.

Table 3. (Processing full-text documents, One-CID, SP-ci) The table shows the predictions from the full-text documents from BioCreAtIve II based on the case-insensitive use of the SP-tagger. All findings are categorized according to the category of the verb form that has been used in the text in conjunction with the mentioned proteins (see methods section). (for use of acronyms see table 2)

	Predictions	Correct predictions	Precision	Recall	F-measure
All, 3-CO	15,823	609	3.8%	28.8%	6.8%
All, SynP	2,078	358	17.2%	17.0%	**17.1%**
Associate, 3-CO	1,203	180	15.0%	8.5%	10.9%
Associate, SynP	171	66	**38.6%**	3.1%	5.8%
MI, 3-CO	1,092	71	6.5%	3.4%	4.4%
MI, SynP	124	23	18.5%	1.1%	2.1%
NMI, 3-CO	14,833	596	4.0%	28.2%	7.0%
NMI, SynP	1,893	325	17.2%	15.4%	**16.2%**

4 DISCUSSION

In the presented work, we defined the classes of modifying interactions containing all verb forms that report on a chemical transformation of one interaction partner (posttranslational modifications, e.g., methylation, acetylation, phosphorylation), and non-modifying interactions (e.g., interaction, binding, regulatory events). The last class is composed of the undefined interactions (e.g., associations, functions). Much to our surprise the single entry from the class of undefined interactions ("associate") contributed significantly to the correct predictions in our analysis. A significant portion of the "association" of protein pairs could be confirmed by a more informative relation between the proteins from the same document.

(Friedman et al., 2001) proposed a categorization of verbs into semantic classes for actions, process and other relations. It is more fine-grained and distinguishes positive regulation ("activate") from negative regulation ("inactivate") and proposes semantic classes related to bond formation ("createbond", "breakbond") and general modification actions, reaction actions and others. This approach shows foresight, but could be too detailed to deliver conclusive results from information extraction.

For the ongoing work in the extraction of gene regulatory events, we will analyze how MI and NMI events contribute to the event extraction.

Acknowledgments

This research was sponsored by the EC STREP project "BOOTStrep" (FP6-028099, www.bootstrep.org) including the development of the Term Repository (the prestage to the BioLexicon). Whatizit has been funded by the NoE "Semantic Mining" (NoE 507505).

References

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.

Friedman,C., et al. (2001) GENIES: a natural-language processing system for the extraction of molecular pathways from journal articles. Bioinformatics, 17(Suppl 1), S74–82.

Gaudan, S., Kirsch, H., and Rebholz-Schuhmann, D. (2005) Resolving abbreviations to their senses in Medline. Bioinformatics 21(18):3658-64

Hakenberg,J., et al. (2005). Systematic feature evaluation for gene name recognition. BMC Bioinformatics; 6 Suppl 1:S9.

Hirschman,L., et al. (2005) Overview of BioCreAtIvE task 1B: normalized gene lists. BMC Bioinformatics, 6(Suppl 1):S11.

Hoffmann,R. and Valencia,A. (2005) Implementing the iHOP concept for navigation of biomedical literature. Bioinformatics 21 (Suppl 2):ii252-8.

Huang,M., et al. (2004) Discovering patterns to extract protein–protein interactions from full texts. Bioinformatics 20(18):3604-3612

Kirsch,H., et al. (2006) Distributed modules for text annotation and IE applied to the biomedical domain. Int. J. Med. Inform. 75(6):496-500.

Krallinger,M., Leitner,F., and Valencia,A. (2007) Assessment of the Second BioCrea-tive PPI task: Automatic Extraction of Protein-Protein Interactions. Proc Second BioCreative Challenge Evaluation Workshop.

Morgan,A., and Hirschman,L. (2007) Overview of BioCreative II Gene normalization. Proc Second BioCreative Challenge Evaluation Workshop.

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. 9;8:50.

Rebholz-Schuhmann,D., et al. (2007a) EBIMed: text crunching to gather facts for proteins from Medline. Bioinformatics 23(2):e237-e244.

Rebholz-Schuhmann,D., et al. (2007b) Text processing through Web services: Calling Whatizit. Bioinformatics 2007 Nov 21

Saric,J., et al. (2006) Extraction of regulatory gene/protein networks from Medline. Bioinformatics 22(6):645-650.

Temkin,J.M. and Gilder,M.R. (2003) Extraction of protein interaction information from unstructured text using a context-free grammar. Bioinformatics. 19(16):2046-53.

Tsuruoka,Y., et al. (2007). Learning string similarity measures for gene/protein name dictionary look-up using logistic regression. Bioinformatics. 23(20):2768-74.

Mining for Gene-Related Key Terms: Where Do We Find Them?

Catalina O. Tudor ⋆ **Carl J. Schmidt** ◦ **K. Vijay-Shanker** ⋆
Department of Computer and Information Sciences ⋆
Department of Animal and Food Sciences ◦
University of Delaware, Newark, DE 19716
tudor@cis.udel.edu schmidtc@udel.edu vijay@cis.udel.edu

Abstract

This paper is concerned about one aspect in the extraction of key terms that describe various types of information about a given gene. Our method for key term extraction is based on a comparison of term occurrences in documents associated with the gene versus a broader set of documents. We investigate the influence on the type of key terms extracted by the type of documents retrieved for the given gene. We provide analysis on five genes to draw our conclusions and hypotheses for future investigations.

1 Introduction

Researchers spend a tremendous amount of time searching the biomedical literature for information they need. A simple PubMed query for a specific gene can sometimes return several thousands of articles, which could be time consuming to read. Instead, we allow researchers to consult a list of most important gene-related information (key terms) gathered automatically from these articles. By consulting key terms and by reading sentences containing a particular key term, the researchers can find quickly information of interest.

For example, searching PubMed for abstracts containing gene *Groucho* returns a list of 269 references to articles. We identify key terms and present users with relevant information: *transcriptional corepressor*, *segmentation*, *neurogenesis* and *WD40*. This immediately informs a user that *Groucho* is a *transcriptional corepressor*, that it might be involved in the processes of *segmentation* and *neurogenesis* and that it might contain the *WD40* domain. From these key terms, researchers can choose to learn more by reading sentences and abstracts containing the terms of interest.

We determine such key terms by comparing the set of documents retrieved for the specific gene (the query set) against a background set of documents with information about genes in general. The type of documents retrieved may influence the type of information captured by the extracted key terms. We investigate how different kinds of key terms can be obtained based on changing the query set. We report our findings about the type of key terms we extracted for five genes when using different query sets. We believe these findings about the influence of the different query sets are not limited to our method for key term extraction, but also to all key term extraction systems that consider term distributions between a background set and a set associated with a given gene.

2 Related Work

One of the earliest works on mining key terms from text is due to Andrade and Valencia (1998). They proposed to automatically mine keywords for families of proteins, by comparing each family's literature against the other families' combined literature. Other systems which also mine key terms from the biomedical literature are built: *e-LiSe* (Gladki et al., 2008), *MedEvi* (Kim et al., 2008), and *Anne O'Tate* (Smalheiser et al., 2008). Our system, eGIFT, **E**xtracting **G**ene **I**nformation **F**rom **T**ext (Tudor et al., 2008), differs from these systems in its intended use only for genes; the construction of background information; the filtering of irrelevant documents; the extension of words to multi-word key terms; the grouping of morphologically related terms; and the division of key terms into categories.

3 Retrieving key terms using eGIFT

We compare the distribution of terms in the abstracts about the gene from some background set. We look for situations where the different frequencies of appearance of a term in two sets of the literature are statistically interesting. For the **Background Set**, we downloaded from PubMed all abstracts for the search on *gene(s)* or *protein(s)*. For the gene-specific documents, we download abstracts from PubMed which mention a given gene name and its synonyms, and call it the **Query Set**. Using these sets of documents we compute the score s_t for a term t as follows:

$$s_t = (\frac{dc_{tq}}{N_q} - \frac{dc_{tb}}{N_b}) * ln\left(\frac{N_b}{dc_{tb}}\right)$$

where dc_{tb} and dc_{tq} are the background and query document counts of t, and N_b and N_q are the total number of documents from the two sets.

The difference between the normalized document frequencies ($\frac{dc_{tq}}{N_q} - \frac{dc_{tb}}{N_b}$) is giving preference to terms that appear more frequently in the Query Set than in the Background Set, while the second part of the equation ($ln\left(\frac{N_b}{dc_{tb}}\right)$) further penalizes common terms in general. We rank the key terms based on their scores, in decreasing order.

4 Research Methods

We have applied our method on 60 genes selected by annotators for a public resource. A set of 5 genes was chosen for our analysis by one of the co-authors expert in Biology and familiar with the selected genes. Their symbols and Entrez Gene IDs are: BMP2 650, GRO 43162, LMO2 4005, OPN 6696, and TERT 7015. Together, we determined the category of each key term, and for each gene we compared the results returned by the different query sets, as described below. For each set, we looked at the top 150 key terms only.

Since the primary goal of this work is to determine how the choice of gene-specific set of documents influences the quality and type of information extracted, we consider for a given gene many different query sets, as will be defined next.

We observed that not all the abstracts from the Query Set are relevant to the given gene. When we search for a specific gene, we obtain two types of abstracts: (1) which talk mainly about the gene, and (2) which are focused primarily on some other topic but happen to mention our gene. Given this observation, we have decided to divide the entire set of retrieved documents for a gene (**Full Set**) into two distinct sets: **About Set** and **Extra Set**. By considering the About Set, instead of the Full Set, we hope to filter out information which is not core to the given gene. We check if an abstract mentions the given gene at least three times, or once in the title, the first or last sentence of the abstract, before assigning it to About Set.

While we expect to obtain more "core" key terms by using About Set as the query set, we also want to see what kind of key terms are found when we use Extra as the query set. However, since Extra documents are supposed to be about some other topic and might just mention our gene, we can focus on the sentences, in the Extra abstracts, that contain our gene, as this might give us gene-related information when mentioned in context of some other topic. So we build a new possible query set, **ExtraSent Set**, that is obtained by taking each document in the Extra Set and only retaining sentences that mention our gene. We similarly obtain **AboutSent** and **FullSent** sets.

Since the title, first and last sentences of the abstracts generally give a high level summary of the work they discuss, we create **AboutTiFL** by only retaining the title, first and last sentences. By using AboutTiFL as the query set, we expect to do well on extraction of high level key terms, but not more detail level key terms [1] , for the gene.

5 Discussion of Results

5.1 About Set vs. Full/Extra Set

As we expected, the use of About as the query set led to better extraction of information that is core to the given gene. For example, processes like *segmentation*, *neurogenesis*, *embryonic development*, and *sex determination* are ranked much higher in the About Set than in the Extra Set for gene *Groucho*. *Groucho* is involved in all of these processes, and since many abstracts "about *Groucho*" will discuss its functions and processes, these terms are highly ranked in contrast to the use of Full Set or Extra Set as the query set. Since the Extra Set abstracts aren't necessarily about *Groucho*, these key terms are ranked much lower and some other key terms take their place in the Extra Set ranking. We found that the highly ranked key terms for the Full Set include terms

[1] By high level we mean process/functional terms, and by detail level terms we mean other genes and domains/motifs

from both About and Extra and the four processes drop in rank, particularly *embryonic development* and *sex determination*. We see several such cases. For example, consider the association of *Lmo2* with *erythropoiesis*. *Lmo2* was originally identified as an oncogenic protein in human t-cell leukemia and later determined to be essential for erythropoiesis (PMID 9520463). *Chromosomal translocations*, *erythropoiesis*, *tumorigenesis*, and *t-cell development* are ranked higher in About than in Full, and, in fact, with the Extra Set the rank dropped considerably. For the gene *Opn*, *secretion*, *cell adhesion*, and *metastasis* ranked very high in the About Set, while only one of these terms ranked in the top 150 key terms for Extra Set.

In contrast, the use of Extra Set as the query set reveals some highly interesting and potentially useful information about the genes which get ranked much lower in the About Set. Rather than high level process/function oriented key terms, with Extra ranking we are able to extract information that is often "lower level", such as other related genes and domains/motifs. Although some of the key terms obtained by using Extra Set are relevant to the given gene, many are "false positives" (i.e. highly ranked terms that were not associated with the gene).

5.2 Sentence-based Document Sets

ExtraSent Set. Extra Set contains many terms that are extraneous to our gene. Hence, we propose to investigate the use of ExtraSent Set as this might filter out terms less relevant to our gene. We notice that this is exactly the situation. Genes and motifs retrieved by using the Extra Set get ranked even better with ExtraSent Set. For example, *eh1* and *bhlh*, which are highly ranked in ExtraSent as compared to About, are domains that are contained in other genes which interact with *Groucho*. Abstracts that focus on other topics/genes but which also mention *Groucho* (and hence make it into Extra Set of *Groucho*) discuss *eh1* and *bhlh* frequently.

Also, some genes are highly ranked with ExtraSent Set when they co-occur frequently with our gene. This might happen when several genes are mentioned together because they form a complex, participate in some pathway, contain a common motif, are expressed in some disease, etc. For example, the gene *Lyl1*, is mentioned by Entrez Gene for interacting with *Lmo2*. ExtraSent is the only set which includes *Lyl1* in the first 150 key terms and ranks it at the top of its list.

Another example is *activin* to be discussed in the context of *Bmp2*. *Activin* is in many ways similar to *Bmp2*, and somebody interested in *Bmp2* would want to know this information. But in particular we believe that the relevance of *activin* can be noted in that some sentences not only discuss similarities, but go on to point out some small but significant differences: "... human CHL2 (hCHL2) protein is secreted and binds activin A, but not BMP-2 ..." (PMID 15094188) and "... BMP-2 and activin A induce PC12 cell neuron differentiation ..." (PMID 8663261). So in some sense, *activin*, while not central to *Bmp2*, may be important to researchers interested in *Bmp2*. *Activin* does not rank highly in the About Set (rank 157), nor in FullSent Set (rank 106), but gets a much higher rank of 25 in ExtraSent Set (while in Extra Set it has rank 90).

A similar example can be noticed with the gene *Opn*. Two genes were boosted in the ExtraSent Set (*DMP-1* and *DSPP*) which were otherwise not present in any of the top 150 key terms for the other sets. *Opn*, *DMP-1* and *DSPP* are SIBLING proteins (small integrin-binding ligand, N-linked glycoproteins) (PMID 16776771). Interestingly, the descriptive terms, like *Glycoprotein*, *integrin-binding*, and *ligand* are all ranked high in the About Set and not present in the Full or Extra sets. Hence we might learn from the About Set that *osteopontin* is a SIBLING protein, but we can learn about other SIBLING proteins, like *DSPP* and *DMP-1* only from ExtraSent Set.

Despite a careful examination, we were not able to find any examples of key terms that were ranked significantly higher in Extra Set as compared to ExtraSent Set. More importantly, Extra Set gave several "false positives" (i.e. several highly ranked terms that were not associated with the gene) as compared to ExtraSent Set. This is in line with our original motivation for considering ExtraSent Set.

AboutSent Set. While ExtraSent was noticeably better than Extra Set, we found that this situation was not replicated when we compared About with AboutSent. In fact, when we compared the ranking of different types of key terms and across genes, the rankings of key terms given by About and AboutSent sets were very similar.

While there are some minor differences in the rankings by About Set and AboutSent Set, there was no noticeable pattern and our conclusion was that these provided very similar quality and type of information. In examining the differences between AboutSent and ExtraSent our observations suggested that there is a parallel to the situation we observed when comparing About with Extra.

FullSent Set. The documents in FullSent Set contain all sentences from the AboutSent and the ExtraSent sets. As we noted earlier, we felt that the About Set and AboutSent were not distinguishable, but the ExtraSent did provide better quality than Extra, as well as a useful but different kind of information from About. Preliminary analysis of the rankings of FullSent does indeed suggest that the advantages of these two sentence based documents were captured.

AboutTiFL Set. The reasons we considered the AboutTiFL Set are as follows: the title usually contains a short, yet concise, summary of the abstract, while the first sentence, as an introduction, together with the last sentence, as a conclusion, contain high level informative terms about the studies reported on the given gene. Thus, as we expected, we obtained most of the high level information related to the gene (such as *corepressor* for *Groucho*, *chromosomal translocation* for *Lmo2*, and *phosphoprotein* for *Opn*) but not highly relevant and detail oriented key terms. For example, *alkaline phosphatase activity* was ranked very low in the AboutTiFL for gene *Bmp2* while it ranked considerably high in the About Set. Similarly, other gene names, such as *osteocalcin* and *alp* which score highly in the About Set, do not appear in the top 150 key terms for the AboutTiFL Set. *WRPW* and *WD40* which are domains related to *Groucho* and extracted from the About Set are ranked low in AboutTiFL.

5.3 Conclusions

We have talked about differences among the Full, About, Extra, ExtraSent and AboutSent sets. We have seen how the Full Set does not distinguish extraneous information from important. By dividing the entire document set into About and Extra sets, we helped separate the two relevant types of information. More importantly, we have shown we can filter highly irrelevant information by considering the ExtraSent Set, which boosts ranks for potential interacting genes, similar or different genes, as well as domains and motifs relevant to the gene in question. We believe further investigation of FullSent versus About and ExtraSent sets is needed in order to determine if the About and ExtraSent sets give the most relevant key terms when used together, or if the FullSent set itself captures the information given by the two sets. On the other hand, if only high-level information is required, then we could restrict our query set to sentences in AboutTiFL.

One of key results of this work is that important concepts/key terms, to be associated with a given gene, can be extracted if we look in the right places for the particular type of concept. And hence, in our opinion, the Full Set (i.e. all abstracts retrieved by searching for a gene) is not the right place to extract key terms, whichever type of key term it is. In this context, we wish to point out that other systems appear to be using Full Set and not distinguish between different ways the gene is mentioned in an abstract.

Evaluating key terms is a challenging task, one of the many reasons being due to the lack of a gold set of terms relevant to specific genes. We are currently conducting an evaluation of key terms retrieved by eGIFT, based on ratings received from biologists, as well as by consulting manually created knowledge bases for genes to identify information which is captured/missed by eGIFT.

References

Miguel A Andrade and Alfonso Valencia. 1998. Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families. *Bioinformatics*, 14(7):600–607.

Arek Gladki, Pawel Siedlecki, Szymon Kaczanowski, and Piotr Zielenkiewicz. 2008. e-LiSe–an online tool for finding needles in the '(Medline) haystack'. *Bioinformatics*, 24(8):1115–1117.

Jung-Jae Kim, Piotr Pezik, and Dietrich Rebholz-Schuhmann. 2008. MedEvi: Retrieving textual evidence of relations between biomedical concepts from Medline. *Bioinformatics*.

Neil R Smalheiser, Wei Zhou, and Vetle I Torvik. 2008. Anne O'Tate: A tool to support user-driven summarization, drill-down and browsing of PubMed search results. *Journal of Biomedical Discovery and Collaboration*, 3(1):2–11.

Catalina O Tudor, K Vijay-Shanker, and Carl J Schmidt. 2008. Mining the Biomedical Literature for Genic Information. In *Proceedings of the Workshop on Current Trends in BioNLP*, pages 28–29. Association for Computational Linguistics.

Improving OCR Performance in Biomedical Literature Retrieval through Preprocessing and Postprocessing

Songhua Xu [1], **James McCusker** [2], **Martin Schultz** [1], **and Michael Krauthammer** [2]
[1]Department of Computer Science, Yale University
51 Prospect Street, New Haven, CT 06520-8285, USA
[2]Department of Pathology & Yale Center for Medical Informatics
300 Cedar Street, New Haven, CT 06510, USA
{songhua.xu, james.mccusker, martin.schultz, michael.krauthammer}@yale.edu

Abstract

Today's information retrieval (IR) techniques are mostly text-based. As a consequence, some types of information are beyond the reach of text-based IR systems, which fail in situations where textual information can not be easily accessed, e.g. textual information in biomedical images and figures. To tackle such situations, we propose to augment IR systems with the ability to perform optical character recognition (OCR). A principal obstacle is the accuracy of the OCR procedure, which is often error-prone. In our work, we introduce some preprocessing and postprocessing techniques for improving the OCR performance. Our preprocessing stage is concerned with separating texts from graphical elements in an image so that the graphics in the image would not affect the performance of OCR, as today's OCR engines are optimized for dealing with documents without graphical elements. Our postprocessing stage is concerned with a context-based OCR result correction. Experimental results show that these preprocessing and postprocessing techniques can consistently improve the performance of biomedical image OCR in terms of either precision or recall.

1 Introduction

In biomedical publications, figures and images often concisely summarize a paper's experimental findings and results. Recent studies have therefore explored the use of images to assist in information retrieval (IR) in biomedicine, mostly based on mining the image caption content. We extend this approach by mining the image text, which refers to the text inside biomedical figures and images. To study the potential of using image text for information retrieval over the biomedical literature, we developed a prototype search engine based on image text search called *Yale Image Finder*, which is publicly available at (*http://kauthammerlab.med.yale.edu/imagefinder*). In a high-level evaluation of image search performance, we demonstrated that the search engine is capable of retrieving a higher number of relevant images compared to querying against the image caption alone (Xu et al., 2008).

An obstacle to the development of a text-based search engine is the accuracy of the OCR procedure, which is often error-prone. In our work, we introduce some preprocessing and postprocessing techniques for improving the OCR performance. Our preprocessing step involves layout analysis to detect and extract text from surrounding graphical elements. As a result, graphical elements do not degrade the performance of the OCR engine, which is optimized for dealing with documents without graphical elements. Our postprocessing step is concerned with performing a context-based OCR result correction. The key idea is to capture the textual context for each biomedical image. We assume that texts within biomedical images are discussed in their textual context, i.e. in the image caption, in the paragraph that discusses the image, or in the paper that features the image. We thus correct the raw image OCR result by matching it to the terms found in its context. Experimental results show that these preprocessing and postprocessing techniques

consistently improve the performance of biomedical image OCR in terms of either precision or recall.

2 A Prototype Biomedical Literature Search Engine Based on Image Text

Prior studies have proposed to use image information, mostly image caption, to assist in biomedical IR (see for example (Hearst et al., 2007)). We extend this idea and propose to facilitate the retrieval of biomedical articles by making the image content accessible to IR systems. This offers several advantages over searching over image captions alone. First, captions may not contain all the textual information that is contained in the images. Second, image texts are usually very specific, allowing for precise matching of images with related images. We implemented a prototype system for image and literature retrieval based on image text. We extract image text through image segmentation and Optical Character Recognition (OCR) in biomedical images. For OCR, we used the Image Analysis toolbox (Document Imaging) that is part of Microsoft Office 2003 Professional. Our system has indexed over 100,000 images from public-access biomedical journal papers. A user can compose an image query by specifying the word(s) he expects to appear inside an image, and optionally in the image caption, or in the associate paper title and abstract. Once the query is submitted, he is presented with images that are relevant to his query (see *http://krauthammerlab.med.yale.edu/imagefinder*).

We have investigated several aspects of our system, including the image text extraction performance (Xu et al., 2008). Our results indicate that on average, only about 30% of image text is contained in the caption of images, and that for queries that contained two ore more search strings, we were able to retrieve 30% to 175% more images compared to searching over caption alone.

3 Preprocessing and Postprocessing Techniques for Improving OCR Performance

Since our new biomedical literature search engine functions through searching image texts, the OCR performance will critically affect the performance of our search engine. Therefore, we introduce a set of preprocessing and postprocessing techniques for improving OCR performance.

The key idea behind our preprocessing step is to provide customized layout analysis over images published in academic journals, using histogram-based image processing techniques (Lienhart and Wernicke, 2002; Wu et al., 1999). The analysis identifies image text elements, and subjects them to OCR. The text extraction is repeated after turning an image 90 degrees, to allow for the capture of vertical image labels.

The key operation in our postprocessing step is to cross-check extracted image text against the context of the images, and to retain image text which is mentioned in its context. Such context-based correction can effectively minimize false positive results, as intensively discussed in prior studies (Kukich, 1992; Ringlstetter et al., 2007). In our current implementation, we work with two types of image context: one is constituted by all the words from the article that features the image, and the other is constituted by the words in the public accessible articles from PubMed Central. We call image text correction based on the former context "article-based correction", and image text correction based on the latter context "corpus-based correction".

In this study, we evaluate these preprocessing and postprocessing steps, either alone or in combination. The goal is to determine the optimal processing pipeline to extract text from biomedical images. We evaluate the following processing options:

Plain-uncorrected option This option uses raw OCR output without any preprocessing or postprocessing.

Plain-corrected option This option uses article-based correction in the postprocessing stage.

Layout-uncorrected option This option uses layout analysis in the preprocessing stage.

Layout-corrected option This option uses layout analysis in the preprocessing stage and article-based correction in the postprocessing stage.

Corpus-plain-corrected option This option uses corpus-based correction in the postprocessing stage.

Corpus-layout-corrected option This option uses layout analysis in the preprocessing stage and corpus-based correction in the postprocessing stage.

High-recall option This option combines the plain-uncorrected option and layout-uncorrected option.

High-precision option This option combines the results from the plain-corrected option, layout-corrected option, corpus-plain-corrected option, and corpus-layout-corrected option.

The latter two options combine the best preprocessing and postprocessing procedures to either retrieve most of the image text content (high-recall option) or to retrieve image text context with the highest amount of precision (high-precision option).

4 Evaluation

To evaluate the effectiveness of our OCR correction techniques, we conducted two evaluations, where we compared OCR-extracted and corrected image text against manually extracted image text. The first evaluation focused on 343 random images whose captions contain the word "survival"; the other evaluation focused on 362 random images whose captions contain the word "apoptosis". Both evaluations covered typical biomedical images, such as graphs, diagrams and experimental results.

In Figure 1, we report the results for all the pre- and postprocessing correction options, and combinations thereof, as discussed in Section 3. We analyze the performance with respect to different word lengths. One reason for doing so is that in the postprocessing stage, our context-based correction methods are less efficient for shorter words. This can be intuitively understood as short text strings, which have been erroneously extracted from images, are more likely to be coincidentally mentioned in the image context. According to these results, we find that context-based image text postprocessing improves precision significantly. We also observe that layout-analysis based preprocessing improves recall, specifically when combined with plain (raw) OCR processing. This can be seen in our high-precision option, where we pool the results of layout-analysis based preprocessing with plain (raw) processing, and apply various context-based post-processing steps. Using this option, we achieve the best overall performance in terms of F-rate. Our high-recall option offers the best performance for retrieving terms that are actually mentioned in biomedical images.

5 Conclusion

In this paper, we introduce preprocessing and post-processing techniques for improving OCR-based image text extraction. We show that a combination of image layout analysis and context-based image text correction is most beneficial for boosting OCR performance over biomedical images.

Acknowledgement

This research has been funded by NLM grant 5K22LM009255. We thank Nam Tran, ThaiBinh Luong, Sebastian Szpakowski, and Pavithra Shivakumar for manually labeling the image text in the "survival" and "apoptosis" image sets.

References

Marti A. Hearst, Anna Divoli, Harendra Guturu, Alex Ksikes, Preslav Nakov, Michael A. Wooldridge, and Jerry Ye. 2007. Biotext search engine: beyond abstract search. *Bioinformatics*, 23(16):2196–2197.

Karen Kukich. 1992. Technique for automatically correcting words in text. *ACM Computing Surveys*, 24(4):377–439.

R. Lienhart and A. Wernicke. 2002. Localizing and segmenting text in images and videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(4):256–268, Apr.

Christoph Ringlstetter, Klaus U. Schulz, and Stoyan Mihov. 2007. Adaptive text correction with web-crawled domain-dependent dictionaries. *ACM Transactions on Speech and Language Processing*, 4(4):1–36.

V. Wu, R. Manmatha, and E.M. Riseman. 1999. Textfinder: an automatic system to detect and recognize text in images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1224–1229, Nov.

S. Xu, J. McCusker, and M. Krauthammer. 2008. Yale image finder (YIF): a new search engine for retrieving biomedical images. *Bioinformatics, Advanced Access*, July.

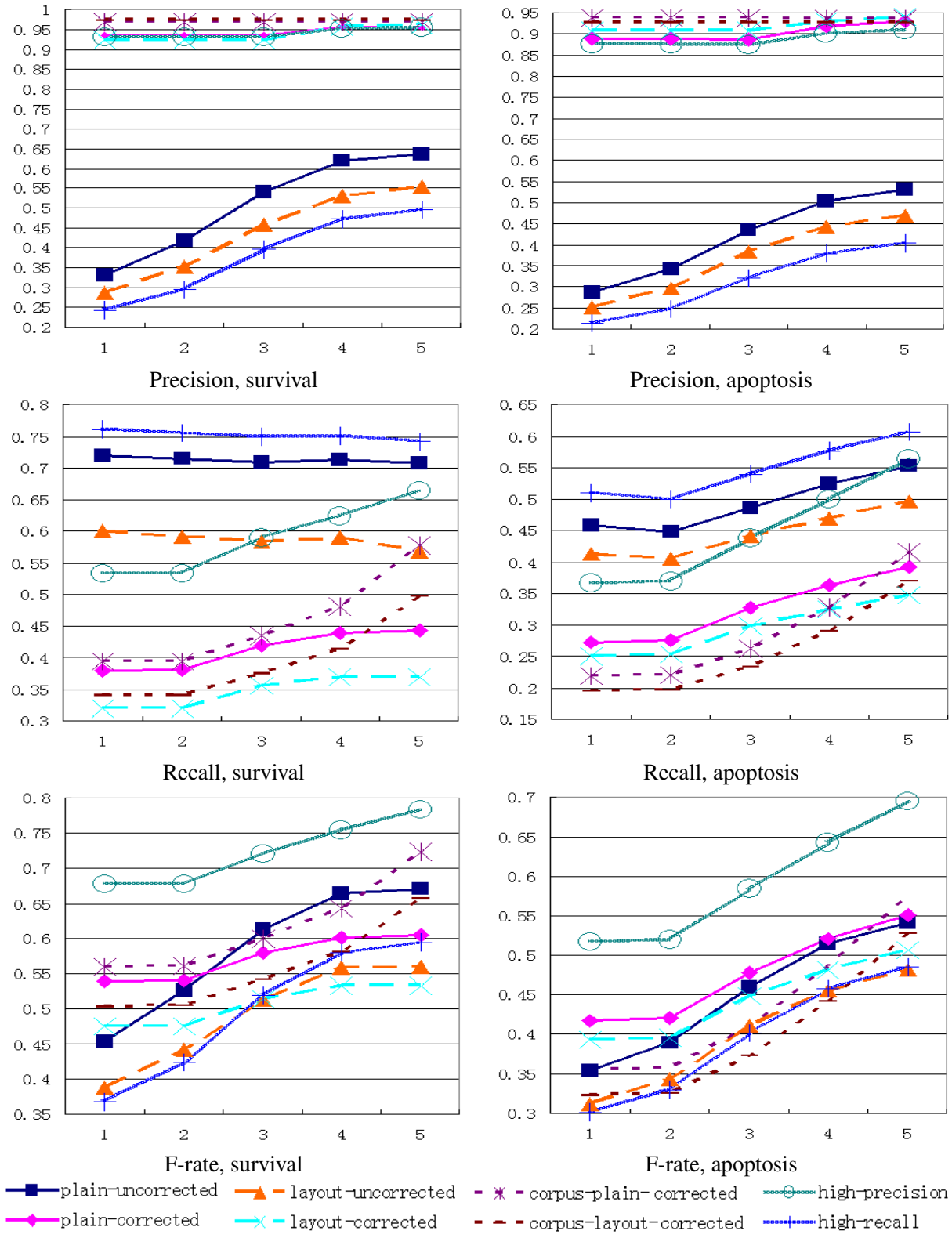


Figure 1: Performance of our method over the survival and apoptosis image sets. Here we show the precision, recall and F-rates (y-axis) for the survival and apoptosis image sets for different pre- and postprocessing methods with respect to different word lengths (x-axis). Results for word length 1 correspond to the overall performance, as we include all words of length 1 and more. From these results, we can see that our high-precision option achieves the best overall performance in terms of F-rate and our high-recall option offers the best performance for retrieving terms that are actually mentioned in biomedical images, i.e. the highest recall.

Work in progress proposals

Towards Ontological Interpretations for Improved Text Mining

Robert Hoehndorf
Research Group *Ontologies in Medicine*, Institute for Medical Informatics, Statistics and Epidemiology and
Department of Computer Science, University of Leipzig and
Department of Evolutionary Genetics, Max-Planck-Institute for Evolutionary Anthropology
hoehndorf@eva.mpg.de

Axel-Cyrille Ngonga Ngomo
Department of Computer Science, University of Leipzig
ngonga@informatik.uni-leipzig.de

Michael Dannemann
Department of Evolutionary Genetics
Max-Planck-Institute for Evolutionary Anthropology
michael_dannemann@eva.mpg.de

1 Motivation

Text mining in biomedicine can be used for several tasks relating both to the extraction of domain-specific knowledge and the management of ontologies. These tasks include the identification of associations between biomedical entities, the extraction of relationships between biomedical entities, the alignment of ontologies and the generation of ontologies from text. Most of the methods used in text mining to perform these tasks are based on statistical measures, algorithms from natural language processing or machine learning. We believe that the overall performance of these methods remains limited as long as no semantic or *ontological* layer is added in the generation and analysis of text mining data. An ontological layer will allow to interpret the results of a text mining analysis with respect to formalized ontological background knowledge, and can be used to generate an *ontological interpretation* of the results of the analysis. In such an ontological interpretation, categories and individuals stand in well-defined ontological relations. The ontological interpretation of text mining results would present several advantages, of which the most important include consistency checks, automated belief revision (ontology curation) and ontologically founded data and information integration.

The generation and analysis of an ontological interpretation of text mining results are not straight forward, as it is necessary to deal both with inconsistent and incomplete knowledge. Classical logics will prove to be insufficient for such a task. Therefore, a non-classical, non-monotonic logic together with non-classical inferences such as abduction and induction is required.

2 Method

For our purpose, text mining identifies references to four kinds of ontological entities in text: categories C, individuals I, relations R and instances of relations T. A category is an intensional entity that can have instances. Instances of categories can be both individuals or other categories. Individuals cannot be instantiated (Herre et al., 2006). A relation such as *instance-of* or *part-of* is an ontological entity that specifies a kind of interaction between multiple entities. Relations have instances that are part of the world. The instances of relations are "the glue that holds things together, the primary constituents of the facts that go to make up reality" (Barwise, 1988). Without loss of generality, we restrict our discussion to binary relations and $R \subseteq (C \cup I) \times (C \cup I)$. We call the structure $\mathcal{TM} =< C, I, R, T >$ resulting from a text mining analysis a *text mining structure* (TMS).

The global aim of the research proposed herein is to provide an ontological interpretation of such a TMS. We apply this interpretation for the refinement of the TMS using the axioms of an ontology. In order to deal with inconsistent and incomplete knowledge, we use a non-monotonic form of logical deduction as a method to consistently generate explanations for facts resulting from this ontological interpretation.

In our work, an ontology is a structure $O =<$

$C', R', ::, isa, Ax >$ of categories C' and relations R' together with a set of axioms Ax.

Definition 1. *An* ontological interpretation I *of a TMS* $\mathcal{TM} =< C, I, R, T >$ *with respect to the ontology* $O =< C', R', ::, isa, Ax >$ *satisfies:*

- *for each* $c \in C$, $c^I = c'$ *such that* $c' \in C'$ *and either* $c :: c'$ *or* $isa(c, c')$,
- *for each* $i \in I$, $i^I = i'$ *such that there exists a* $c' \in C'$ *and* $i :: c'$,
- *for each* $r \in R$, $r^I = r'$ *such that* $r' \in R'$ *and* $isa(r, r')$,
- *for each* $t \in T$, $t^I = t'$ *such that there exists a* $r' \in R'$ *and* $t' :: r'$.

An ontological interpretation performs the following functions: for each category identified in the text, it identifies at least one category in the ontology O of which the category found in the text is either a sub-category or an instance; for each individual in the text, it identifies at least one category of which this individual is an instance; and similarly for relations and their instances.

Two major difficulties arise when trying to find an ontological interpretation of a TMS. First, it may occur that no ontological interpretation exists due to an inconsistency. In this case, we call the TMS $\mathcal{TM}$ classically inconsistent with the ontology O. Second, there may be many possible ontological interpretations for a TMS, and some measure of preference should be established to select the most appropriate ontological interpretation.

In order to deal with inconsistencies, we attempt to establish classical consistency by extending the ontological interpretation such that identified categories (or instances) are subclasses (or instances) of more general categories. For example, consider a TMS containing the following three relation instances:

$$IsA(Arsenic, Poison) \quad (1)$$

$$PlaysRole(Arsenic, Poison) \quad (2)$$

$$HasFunction(Arsenic, Poison) \quad (3)$$

Here, poison is used in three mutually exclusive meanings: as a substance, a role and a function; any ontological interpretation interpreting *Poison*, *IsA*, *PlaysRole* and *HasFunction* in their usual understanding will be classically inconsistent. Interpreting *Poison* as a subclass of *Entity* avoids the inconsistency, but does not permit inferences based on axioms pertaining to more specific categories. Abductive reasoning can be used to fill the gap: abduction is a non-classical form of inference that generates a *minimal* explanation for an observation. The general schema for abduction is: $B, A \rightarrow B \vdash A$. As an assumption, we use the following formula, where C_i ranges over all categories from O:

$$isa(Poison, C_1) \vee \ldots \vee isa(Poison, C_n) \rightarrow isa(Poison, Entity) \quad (4)$$

Abduction can then generate the desired and consistent minimal explanation for (4)

$$isa(Poison, Substance) \vee isa(Poison, Role) \vee isa(Poison, Function) \quad (5)$$

3 Conclusion

We suggest that ontological interpretations can improve text mining results by providing an additional semantic structuring layer. This layer can be used to disambiguate the kind of relations and categories identified through text mining, and to identify categories of which recognized named entities are instances. Formal ontologies play a crucial role in this step. The use of abductive reasoning can lead to rich and consistent ontological interpretations that contain explanations for the facts identified through text mining. These explanations can be used subsequently for the identification of novel hypotheses or the integration of knowledge. Ultimately, using ontological interpretations provides a starting point for elevating the results of text mining analyses from data to knowledge.

References

John Barwise. 1988. *The situation in logic*. CSLI Publications.

H. Herre, B. Heller, P. Burek, R. Hoehndorf, F. Loebe, and H. Michalek. 2006. General Formal Ontology (GFO) – A foundational ontology integrating objects and processes [Version 1.0]. Onto-Med Report 8, Research Group Ontologies in Medicine, Institute of Medical Informatics, Statistics and Epidemiology, University of Leipzig, Leipzig, Germany.

Towards Standardisation of Named-entity Annotations in the Life Science Literature

Dietrich Rebholz-Schuhmann[1] and **Goran Nenadic**[2]
[1]European Bioinformatics Institute
Wellcome Trust Genome Campus, Hinxton, Cambridge, CB10 1SD, U.K
[2] School of Computer Science, University of Manchester
Oxford Road, Manchester, M13 9PL, U.K.
rebholz@ebi.ac.uk, G.Nenadic@manchester.ac.uk

Summary

The main aim of this proposal is to revisit and in the best case re-launch an initiative that would provide harmonised ways for representing and tagging named entities in the life science literature. We are proposing to establish common document formats that facilitate the exchange of annotation *results* contained in the literature as a complementary approach to the development of interoperable *tools*. We want to work towards (a) recommendations for a common syntax to embody entity mentions in publishers' document formats (e.g., into PMC), and (b) provision of a common way to reference semantic types. The main stakeholders (text mining users, researchers, service providers and publishers) would need to build an infrastructure that integrates literature resources with entity databases. The main benefits result from better integration of literature resources and text-mining results with data from other biomedical research groups and from the identification of the next generation challenges for novel text mining research.

1 Motivation, aims and stakeholders

Identification and annotation of entities of different semantic types is the key factor for accessing biomedical literature. While there have been numerous solutions proposed to identify entities in text (see BioCreAtIve initiative), there are very few community-wide efforts to provide harmonised annotations both for the syntactic and semantic levels, which would facilitate interoperability and re-use of processed documents (Krallinger et al., 2007). This is in contrast to widespread attempts to standardise semantic descriptions and exchange of non-textual biomedical data. Instead, text mining solutions are typically based on their own annotation schemas, making it difficult for the community to easily combine and expand different solutions. This also hinders further developments in the area, as many user and research groups need to allocate significant resources in re-developing and re-aligning existing solutions.

We would therefore like to re-launch an initiative that would result in a community-agreed way for representing and tagging named entities (NEs) in biomedical documents. A harmonised approach would provide the stakeholders with the following:

- the users would be able to use annotated results from different sites (i.e., repositories) to have efficient knowledge acquisition and exploitation (e.g., semantics-based browsing, visualisation, integration);
- the text mining research and service provision communities would profit from document annotations originated from different applications to improve the state of the art in NER, and motivate progress in other text mining tasks;
- publishers and industry would be able to provide an added value to their products, and thus facilitate data sharing, availability and interoperability.

2 Harmonising annotation of named entities: needs and obstacles

Informal discussions within the bio-text mining community (Kevin Cohen, BioNLP) have concluded that more efforts are needed to provide interoperability of tools and data, and — in partic-

ular — that named entities would make an optimal level for text annotations that would facilitate the exchange of text mining results. Recent initiatives from publishers (e.g. Elsevier, FEBS Letter experiment) have re- affirmed these conclusions: both users and data providers are interested in "changing the ways science is published" (the Elsevier Grand challenge[1] 2008), and it seems that annotating and linking NEs to databases is a minimal requirement to support this aim. Publishers already consider requesting authors to annotate key entities in their articles (at least at the document level). Although there are still issues in bio-NER, it would be useful to enable users and developers alike to move beyond named entity recognition by providing documents with pre-annotated NEs in a common format, so that they can use pre-calculated NE annotations for visualisation, browsing, indexing or further processing. Many applications need NEs recognised before any further processing, and a common way of their annotation would only improve the possibility for using and sharing results, as well as for improving research that depends on NE annotations.

The main obstacles in this process are that several research and service provision groups have already developed and used numerous in-house formats and that there is no theoretical consensus on certain annotation issues (e.g. representation of ambiguities). There have been several attempts to address representation of NEs in the community (e.g. IeXML, SciXML, Genia, TXM, Termino, etc.), but to the best of our knowledge, so far there is not a comprehensive comparative analysis between different (text-mining derived) annotation schemas. Furthermore, there have been very few attempts to integrate publisher/archiving annotation formats with text mining results (e.g. IeXML, partly SciXML, Genia) (Rebholz-Schuhmann et al., 2006; Copestake et al., 2006; Kim et al., 2003; Harkema et al., 2005)

Data representation that supports interaction with end users (both experts and non-experts) has also been identified as one of the key objectives of the recently launched EU Elixir project[2], which aims to examine the status of literature repositories throughout Europe and provide recommendations for a future information-sharing infrastructure platform that would integrate databases and literature.

3 Proposed approach

We would like to design a minimal tag set that would be *integrated* into publishers' formats and be part of meta-data used to annotate NE mentions in text and point to their semantic types and their referent IDs (if available). We would like to develop an industry-wide solution that would make interoperability much more realistic. In addition to syntactic harmonisation, we would also like to discuss semantic "normalisation" and a common way to point to (external) semantic resources. More precisely, we would like to initiate further discussions on the harmonisation of representations of bio-NEs in documents, including:

- at the syntactic level, the identification of a minimal set of NE tags and features (inline and stand-off) to be included in publishers' formats, including representation of ambiguities and multiple annotations (e.g. annotations from different groups/services);
- at the semantic level: the integration of a basic semantic type system into document formats, including the provisions for using references/pointers to external type systems (e.g. existing ontologies or purposely-built type systems[3]).

A solution would be to (a) implement a common basic/minimal syntax to annotate entity mentions in documents, and (b) provide a common way to point to (potentially external) semantic types. This way we would provide data exchange and interoperability on the level of data (in addition to potential interoperability of tools).

4 Road map

One of the results from previous discussions was a minimal annotation framework that included a single tag and number of mandatory (semantic) attributes describing entities (Rebholz-Schuhmann et al., 2006). Building on that as well as other contributions, we suggest the following road map:

1. Discuss and identify during the discussion at the SMBM 2008 the potential benefits and

[1] http://www.elseviergrandchallenge.com/

[2] http://www.elixir-europe.org

[3] E.g. a UIMA complaint type system at: http://www.u-compare.org/type_system.html

obstacles as well as issues of shared and disjoint interest.

2. Identify a working group to prepare a set of recommendations, following the consultations with interested research groups, publishers, service provides (e.g. EBI, NaCTeM, BioCreative Meta-server, etc.) and organisers of text mining challenges (e.g. BioCreative). The group will recommend a minimal annotation type system and invite for comments from the community and stakeholders.

References

A. Copestake, P. Corbett, P. Murray-Rust, C. J. Rupp, A. Siddharthan, S. Teufel, and B. Waldron. 2006. An architecture for language processing for scientific texts. In *Proc. UK e-Science All Hands Meeting*.

H. Harkema, I. Roberts, R. Gaizauskas, and M. Hepple. 2005. A web service for biomedical term lookup. *Comparative and Functional Genomics*.

J.-D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii. 2003. GENIA corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19:i180–182.

M. Krallinger, F. Leitner, and A. Valencia. 2007. Assessment of the second BioCreative PPI task: Automatic extraction of protein-protein interactions. In *Proc. BioCreative II*.

D. Rebholz-Schuhmann, H. Kirsch, and G. Nenadic. 2006. IeXML: towards an annotation framework for biomedical semantic types enabling interoperability of text processing modules. In *Proc. BioLink, ISMB 2006*.

Author index

Turku Centre for Computer Science
TUCS General Publications

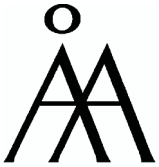
29. **João M. Fernandes, Johan Lilius, Ricardo J. Machado and Ivan Porres (Eds.)**, Proceedings of the 1st International Workshop on Model-Based Methodologies for Pervasive and Embedded Software
30. **Mats Aspnäs, Christel Donner, Monika Eklund, Ulrika Gustafsson, Timo Järvi and Nina Kivinen (Eds.)**, Turku Centre for Computer Science, Annual Report 2003
31. **Andrei Sabelfeld (Editor)**, Foundations of Computer Security
32. **Eugen Czeizler and Jarkko Kari (Eds.)**, Proceedings of the Workshop on Discrete Models for Complex Systems
33. **Peter Selinger (Editor)**, Proceedings of the 2nd International Workshop on Quantum Programming Languages
34. **Kai Koskimies, Johan Lilius, Ivan Porres and Kasper Østerbye (Eds.)**, Proceedings of the 11th Nordic Workshop on Programming and Software Development Tools and Techniques, NWPER'2004
35. **Kai Koskimies, Ludwik Kuzniarz, Johan Lilius and Ivan Porres (Eds.)**, Proceedings of the 2nd Nordic Workshop on the Unified Modeling Language, NWUML'2004
36. **Franca Cantoni and Hannu Salmela (Eds.)**, Proceedings of the Finnish-Italian Workshop on Information Systems, FIWIS 2004
37. **Ralph-Johan Back and Kaisa Sere**, CREST Progress Report 2002-2003
38. **Mats Aspnäs, Christel Donner, Monika Eklund, Ulrika Gustafsson, Timo Järvi and Nina Kivinen (Eds.)**, Turku Centre for Computer Science, Annual Report 2004
39. **Johan Lilius, Ricardo J. Machado, Dragos Truscan and João M. Fernandes (Eds.)**, Proceedings of MOMPES'05, 2nd International Workshop on Model-Based Methodologies for Pervasive and Embedded Software
40. **Ralph-Johan Back, Kaisa Sere and Luigia Petre**, CREST Progress Report 2004-2005
41. **Tapio Salakoski, Tomi Mäntylä and Mikko Laakso (Eds.)**, Koli Calling 2005 - Proceedings of the Fifth Koli Calling Conference on Computer Science Education
42. **Petri Paju, Nina Kivinen, Timo Järvi and Jouko Ruissalo (Eds.)**, History of Nordic Computing - HiNC2
43. **Tero Harju and Juhani Karhumäki (Eds.)**, Proceedings of the Workshop on Fibonacci Words 2006
44. **Michal Kunc and Alexander Okhotin (Eds.)**, Theory and Applications of Language Equations, Proceedings of the 1st International Workshop, Turku, Finland, 2 July 2007
45. **Mika Hirvensalo, Vesa Halava and Igor Potapov, Jarkko Kari (Eds.)**, Proceedings of the Satellite Workshops of DLT 2007
46. **Anne-Maria Ernvall-Hytönen, Matti, Jutila, Juhani Karhumäki and Arto Lepistö (Eds.)**, Proceedings of Conference on Algorithmic Number Theory 2007
47. **Ralph-Johan Back and Ion Petre (Eds.)**, Proceedings of COMPMOD 2008
48. **Elena Troubitsyna (Editor)**, Proceedings of Doctoral Symposium held in conjunction with Formal Methods 2008
49. **Reima Suomi and Sanna Apiainen (Eds.)**, Promoting Health in Urban Living: Proceedings of the Second International Conference on Well-being in the Information Society (WIS 2008)
50. **Aulis Tuominen, Jussi Kantola, Arho Suominen and Sami Hyrynsalmi (Eds.)**, NEXT 2008 - Proceedings of the Fifth International New Exploratory Technologies Conference
51. **Tapio Salakoski, Dietrich Rebholz-Schuhmann and Sampo Pyysalo (Eds.)**, Proceedings of the Third International Symposium on Semantic Mining in Biomedicine (SMBM 2008)

Turku
Centre *for*
Computer
Science

Joukahaisenkatu 3-5 B, 20520 Turku, Finland | www.tucs.fi

University of Turku

- Department of Information Technology
- Department of Mathematics



Åbo Akademi University

- Department of Information Technologies

Turku School of Economics

- Institute of Information Systems Sciences

ISBN 978-952-12-2133-0
ISSN 1239-1905

Proceedings of SMBM 2008

Proceedings of SMBM 2008

Proceedings of SMBM 2008