Enhanced semantic access to the protein engineering literature using ontologies populated by text mining

René Witte* and Thomas Kappler

Institute for Program Structures and Data Organization (IPD),
Faculty of Informatics, Universität Karlsruhe (TH), Germany;
Email: {witte|kappler}@ipd.uka.de
*Corresponding author

Christopher J. O. Baker

Data Mining Department, Computing Division
Institute for Infocomm Research (I^2R), Singapore
Email: cbaker@i2r.a-star.edu.sg

Abstract: The biomedical literature is growing at an ever-increasing rate, which pronounces the need to support scientists with advanced, automated means of accessing knowledge. We investigate a novel approach employing description logics (DL)-based queries made to formal ontologies that have been created using the results of text mining full-text research papers. In this paradigm, an OWL-DL ontology becomes populated with instances detected through natural language processing (NLP). The generated ontology can be queried by biologists using DL reasoners or integrated into bioinformatics workflows for further automated analyses. We demonstrate the feasibility of this approach with a system targeting the protein mutation literature.

Keywords: text mining; semantic web; ontological NLP; protein mutations; automated reasoning in bioinformatics; querying OWL-DL ontologies; description logics.

Reference to this paper should be made as follows: Witte, R., Kappler, T. and Baker, C.J.O. (2007) 'Enhanced semantic access to the protein engineering literature using ontologies populated by text mining', *Int. J. Bioinformatics Research and Applications*, Vol. 3, No. 3, pp. 389–413.

Biographical notes: René Witte is currently a research associate at the IPD, University of Karlsruhe, Germany, where he heads the Text Mining group. His work encompasses foundations in computational linguistics, software engineering aspects of natural language processing systems, as well as the application of text mining in diverse areas such as news analysis, building architecture, biomedical research, and software engineering. He previously worked as a post-doctoral fellow and research associate within the computational linguistics and software engineering groups, both at Concordia University in Montréal, Canada. Dr. Witte holds a Doctorate of Engineering (Dr.-Ing.) and a Diploma in Informatics from the University of Karlsruhe.

Thomas Kappler is currently a graduate student of Informatics at the University of Karlsruhe, Germany. His work at the IPD includes biological text mining,

as well as developing interfaces and semantic extensions for MediaWiki. Other research interests include software engineering with a focus on formal techniques and performance modeling.

Christopher J. O. Baker is a principle investigator in the Knowledge Discovery Department, Institute for Infocomm Research (I^2R), Singapore. His research interests include the application of Semantic Web technologies in the life sciences. Dr. Baker was formerly bioinformatics project manager for the Génome Québec project: Ontologies, the Semantic Web and Intelligent Systems for Genomics. He was also group leader In-silico Discovery at Ecopia BioSciences where he managed the annotation team and masterminded key portions of the DecipherIT bioinformatics suite. Dr. Baker co-edited the volume 'Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences'. He holds a PhD from the University of Wales, Cardiff, UK.

1 Introduction

A large amount of today's biological knowledge is only available from full-text research papers. Since neither manual database curators nor users can keep up with the rapidly expanding volume of scientific literature, natural language processing (NLP) approaches are becoming increasingly important for bioinformatics projects. *Text Mining* systems for biology and biomedicine apply specific NLP techniques to analyse, extract, and cross-link information detected in abstracts or full-text research papers (Ananiadou and McNaught, 2006; Cohen and Hersh, 2005; Rebholz-Schuhmann et al., 2005). *Ontology* is an important technology for modeling information within a domain and many biological ontologies have been developed to facilitate structuring, sharing, and accessing biological knowledge (Bodenreider, 2006; Lambrix et al., 2007). Biological text mining systems often incorporate ontological knowledge bases, for example, as a language processing resource or result export format.

In this paper, we investigate a particular application of biological ontologies in combination with text mining: An ontology for a particular sub-domain is modeled for use with a language processing system and subsequently populated with instances detected in either abstracts or full-text papers. The resulting ontology is then delivered to the user in a standardized format (namely, the *Web Ontology Language*, OWL-DL) for querying. Given that a multitude of specific text segments are generated when text mining a large body of scientific literature, querying the ontology is the equivalent of interrogating a summary of the whole domain of discourse, saving significant time in finding and reading relevant literature. This may in turn lead scientists to adopt a new approach to information retrieval, which is cross-platform and content-specific rather than document-centric. Accessing the full text of a paper may become a secondary step occurring after the query of entity-specific text segments or tiles from an NLP-instantiated ontology, invoked effortlessly from a user's desktop. A biologist could choose to examine the text of the identified documents and/or sentences, query another (NLP-)populated ontology, or opt to establish automated workflows for bioinformatic annotation pipelines.

Paper outline. This paper is structured as follows: In the next section, we describe our text mining system and its application domain, protein engineering literature. Section 3 details the design, implementation, and initialization of an ontology supporting this domain.

The following Section 4 describes the population of this ontology from the literature using the ontology-extended text mining system. How the result ontology can then be queried by biologists is detailed in Section 5. The evaluation of our system is presented in Section 6, followed by a discussion and related work in Section 7. Finally, conclusions and future work are outlined in Section 8.

2 Protein Engineering Literature and Mutation Miner

For protein engineers, understanding the impact of all mutations carried out on a protein family requires a complex mapping of sequence mutants to a common structure. Concurrent access to protein structure visualisations and annotations describing the impacts of mutations is possible using the *Protein Mutant Database* (PMD).[a] The content of this database is limited, however, by the speed at which newly published papers can be curated: In 1999, the PMD authors already reported a three-year backlog of unprocessed publications (Kawabata et al., 1999). Thus, there exists a pronounced need to speed up the extraction of mutation-impact information from the scientific literature and make it more readily available to protein engineers. This has been our motivation for designing a text mining system capable of analysing enzyme mutation experiments described in full-text research papers: *Mutation Miner*, which we describe in the following subsection.

2.1 Mutation Miner

The goal of this work is the annotation of 3D protein structures with segments of literature detailing the consequences of specific mutations. Mutation Miner (Baker and Witte, 2006; Witte and Baker, 2005) is a sophisticated information system designed for this purpose that comprises an initial stage text mining subsystem linked to subsequent protein sequence retrieval and analysis subsystems. With Mutation Miner, a protein engineer can view structural representations of proteins (obtained from protein databases) combined with annotations describing mutations and their impacts (extracted through text mining from publications) within a unified visualisation using a tool like ProSAT (Gabdoulline et al., 2006).

The natural language analysis subsystem has been developed based on the GATE *(General Architecture for Text Engineering)* framework (Cunningham et al., 2002). GATE is a component-based architecture, where documents are processed through *pipelines* of NLP components. This permits the dynamical assembly of a text mining application through adding, swapping, or re-ordering its components. Several standard components are supplied with the architecture, like a part-of-speech (POS) tagger, a gazetteer that assigns semantic labels to tokens (words) in a text, and the JAPE language (Cunningham et al., 2000) for expressing grammar rules, which are compiled into finite-state transducers. Results are exchanged between the components through document *annotations* using a form of stand-off markup. For further technical details on GATE, we refer the reader to the online documentation.[b]

[a] Protein Mutant Database (PMD), `http://pmd.ddbj.nig.ac.jp/`
[b] GATE documentation, `http://gate.ac.uk/documentation.html`

2.2 Ontology Extensions

Mutation Miner was originally developed without innate support for ontologies: Resources were converted from external formats (like databases or taxonomies) into structures supported by GATE (like gazetteer lists).

In this paper, we describe our most recent work on an almost complete re-engineering of the NLP pipeline for the support of biological ontologies. These extensions are twofold; Firstly, we aim to replace individual language resources, which are used by the different language analysis components and come in a multitude of formats, with a single, unified representation: an ontology in OWL-DL format. Secondly, we also added capabilities for exporting text mining results in OWL-DL. This facilitates the deployment of *Onto Mutation Miner* in a Semantic Web context: Biologist can now query the result ontology directly or integrate it within a larger bioinformatics workflow.

3 Ontology Design and Initialization

Ontologies explicitly represent domains using entities, properties, and relationships. They can serve as a common vocabulary enabling semantic interoperability between information systems, databases, and users in a multitude of domains (Obrst et al., 2007), leading to a better understanding of the field, as well as more effective and efficient handling of information.

In concert with the adoption of the Gene Ontology (GO),[c] biologists have come to recognise ontologies as vehicles intended to capture and represent aspects of the real world, providing an essential technology for addressing the challenges of the post-genomics era. Ontologies are increasingly made available in standardised formats, such as OWL (Smith et al., 2004), which can be processed by a variety of tools and systems. Some biological ontologies (see OBO[d]) have been engineered to a high level of maturity and stability with respect to knowledge representation, while others, such as the FungalWeb Ontology (Shaban-Nejad et al., 2005), have started out immediately with the goal of providing a knowledge base containing instances.

Our mutation ontology differs from typical biological ontologies in that it forms the central data structure and knowledge base for a text mining system. As discussed in (Witte et al., 2007), it therefore needs additional information for supporting the various natural language tasks (see Section 4). Additionally, we provide for the instantiation of textual entities (e.g., sentences) together with the biological entities they describe. Our design is covered in Section 3.1.

Before the modeled ontology can serve as a knowledge base, it needs to be initialized with information concerning the biological entities under consideration, in particular proteins and organisms. For this task, we developed a novel instantiation approach that accesses and integrates various publicly available databases into our ontology, as described in Section 3.2.

3.1 Ontology Design

An ontology that can house instances from Mutation Miner requires concepts for the main units of discourse—proteins, mutations, organisms—as well as supplementary concepts that

[c] The Gene Ontology, `http://www.geneontology.org/`
[d] Open Biological Ontologies (OBO), `http://obo.sourceforge.net/`

Table 1 Ontological concept definitions and instance examples for Mutation Miner

Concept	Definition	Example Instances
Cellular Component	Subcellular structures, locations, and macromolecular complexes	Ribosome, Golgi, Vesicle
Protein	A complex natural substance that has a high molecular weight and a globular or fibrous structure composed of amino acids linked by peptide bonds	Protein, Immunoglobulin
Organism	A virus or a unicellular or multicellular prokaryote or eukaryote	S. lividans, Clostridium thermocellum
Enzyme	A protein that acts as a catalyst, speeding the rate at which a biochemical reaction proceeds but not altering the nature of the reaction	Xylanase A, endo-1,4-β-xylanase
Recombinant Enzyme	Enzymes produced from new combinations of DNA fragments using molecular biology techniques	Xylanase A+E210D
Mutant	Indicates that something is produced by or follows a mutation; also a mutant gene or protein	E210D, Phe37Ala, Arg115
Measurement	Units of measurement	half life (s), Kcat, pH, hydrolysis efficiency
Property	The description of a biological, chemical or physical property of a protein that can be quantified	denaturation, catalysis, stabilization, unfolding
Impact	An examination of two or more enzymes (wild type or mutant) to establish similarities and dissimilarities	shift, increase, more active, fold, destabilize

characterize changes in enzyme properties, the direction of the change, and the biological property of the enzyme that has been altered. Table 1 shows the main concepts together with a brief definition and Figure 1 shows a part of the ontology graphically.

The ontology is represented in OWL-DL and was created using the Protégé-OWL extension of Protégé,[e] a free ontology editor. Here, we made use of two OWL language elements that model important information about the domain. Firstly, using *object properties*,[f] which specify relations between class instances, we register several relationships between instances of ontology classes. For example, the *Mutation* class has a *changedGene* object property, which is defined as having the domain "Mutation" and the range "Gene," linking a mutation instance to the instance of the gene it modifies. Secondly, cardinality restrictions are included to model the possible alternatives for denoting an organism. For example, the organism description in a text may consist of at most one genus, species, and strain, respectively, where strain is optional but only if both genus and species are given.

Several other enhancements to the ontology's expressiveness are possible, like placing additional restrictions on relations. They are not necessary, however, for the ontology-enhanced NLP analysis, but could be added to improve reasoning over extracted entities, e.g., for advanced querying.

3.2 Ontology Initialization

Before the ontology can be deployed in an NLP system, instances for the various classes like *protein* or *organism* need to be created. Since adding and maintaining these instances

[e] Protégé ontology editor, `http://protege.stanford.edu/`

[f] OWL Web Ontology Language Guide, Object Properties, `http://www.w3.org/TR/2004/REC-owl-guide-20040210/\#SimpleProperties`

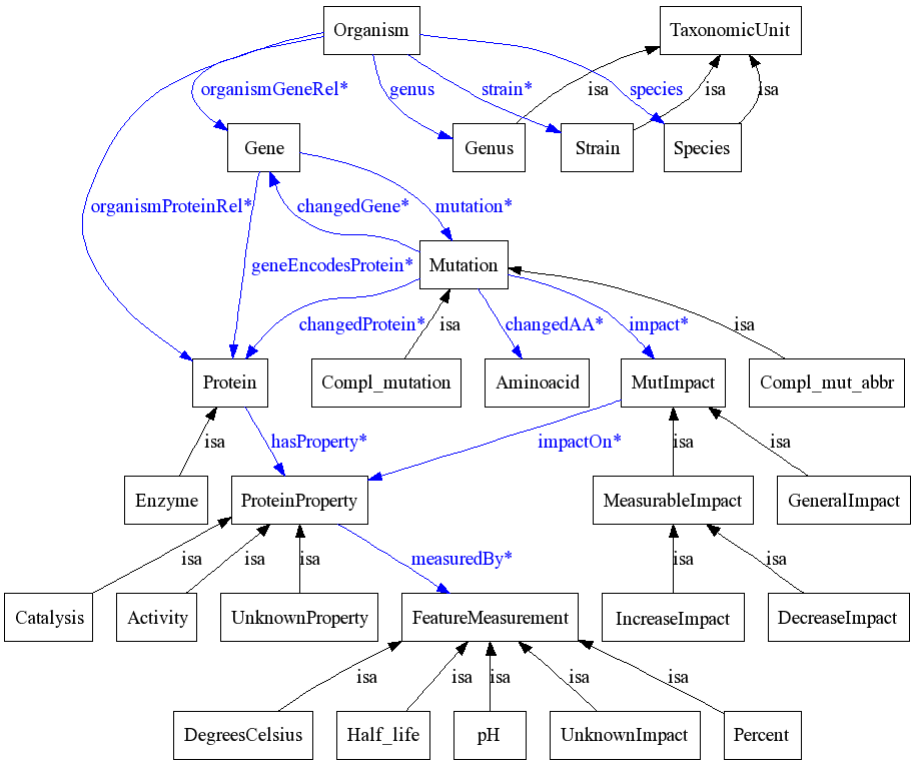

Figure 1 A part of the Mutation Miner ontology

and their relations manually is not an option, we now show how ontology instances can be automatically created and updated with respect to external biological databases.

Organism Instance Initialization. Our text mining system needs to (i) detect organism descriptions in natural language texts, (ii) normalize those descriptions (e.g., in case of abbreviations), and (iii) uniquely ground each detected entity to an external database. Hence, the information required for each of these steps needs to be encoded in the ontology.

The systematic classification of organisms is called *taxonomy*. The individual species are set in relation to each other according to the degree of their genetic relationship. The names of organisms consist of parts called *taxonomic units*, giving the position in the classification tree. Usually, the taxonomic units *genus* and *species* are used in biomedical texts, resulting in a name such as *Escherichia coli*. Sometimes a *strain* is also given, which designates a more precise identification.

We use the *Taxonomy database* (Federhen, 2003) from NCBI[g] to initialize the organism part of our ontology. The Taxonomy database is "a curated set of names and classifications for all of the organisms that are represented in GenBank" (see (Federhen, 2003) for a detailed description). GenBank[h] is another NCBI database, containing "publicly available DNA sequences for more than 165,000 named organisms." As of 2006-06-05, the Taxonomy database contained 310,756 classified taxa, with 409,683 different names in total.

[g]NCBI Taxonomy Homepage, `http://www.ncbi.nlm.nih.gov/Taxonomy/`
[h]GenBank sequence database, `http://www.ncbi.nih.gov/Genbank/index.html`

Table 2 The NCBI Taxonomy entry for E. coli (`tax_id` 562, `rank="species"`)

name_txt	**name_class**
"Bacillus coli" Migula 1895	synonym
"Bacterium coli commune" Escherich 1885	synonym
"Bacterium coli" (Migula 1895) Lehmann and Neumann 1896	synonym
Bacillus coli	synonym
Bacterium coli	synonym
Bacterium coli commune	synonym
Escherchia coli	misspelling
Escherichia coli	scientific name
Escherichia coli (Migula 1895) Castellani and Chalmers 1919	synonym
Eschericia coli	misspelling

In NCBI's database, every species and taxonomic unit has exactly one entry with a name classified as *scientific name,* as well as other possible variants. The scientific name is the "correct" one, and the others can be synonyms, common misspellings, or past names if the organism has been reclassified. Table 2 shows an example entry, constricted to the most important columns, for the organism *Escherichia coli (E. coli)*. It can be seen that there are seven synonyms and two common misspellings recorded in addition to the scientific name.

In order to support named entity detection of organisms, the ontology must contain the taxonomical names so that they can be matched against words in a text using a gazetteer NLP component. This information can also be directly extracted from the NCBI database, including the names themselves and information like the hierarchical structure of taxa and organisms.

Together with the taxonomical information we store additional metadata, like the originating database and the "scientific name," for each instance. This becomes important when delivering provenance information to scientists working with the populated ontology. An additional advantage of replacing flat organism lists with an ontology is that the taxonomical hierarchy is directly represented and can be queried, e.g., by grammar rules. An example for this is given in Section 4.1.

To convert the taxonomy data, it is possible to download the whole database, which is available as structured plain text files from NCBI's FTP server. A Python program was developed for this purpose, which reads these files and inserts their contents into an SQL database, preserving the structure by directly mapping each file to a database table and its columns to SQL columns in that table. The Mutation Miner ontology is subsequently initialized from the contents of this database with a custom Java program using the *Jena* library. Jena[i] is an open source "Semantic Web Framework for Java," providing an API for OWL generation. The resulting comprehensive set of instances can be queried by all language processing components through GATE's ontology layer (Bontcheva et al., 2004).

Protein Instance Initialization. Like for organisms, we need instance information supporting the detection, normalization, and grounding of protein entities in our ontology. The *UniProt Knowledge Base* (Bairoch et al., 2005) is a set of two protein databases, *Swiss-Prot*[j] and *TrEMBL*. Both hold entries about proteins appearing in published works, including information about protein functions, their domain structure, associated organisms, post-translational modifications, variants, among others. Swiss-Prot, which consisted of

[i] Jena, `http://jena.sourceforge.net/`

[j] Swiss-Prot protein database, `http://www.expasy.org/sprot/`

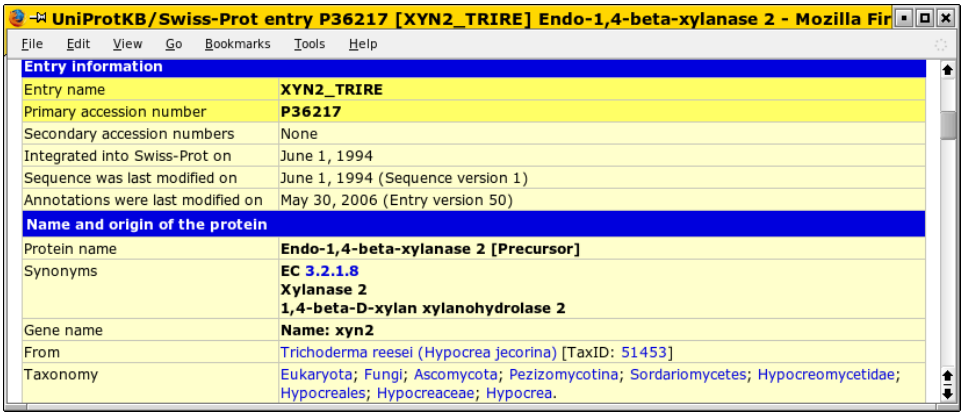

Figure 2 Swiss-Prot entry for *Endo-1,4-beta-xylanase 2*

228,670 entries as of 2006-07-02, contains "manually-annotated records with information extracted from literature and curator-evaluated computational analysis,"[k] while TrEMBL is populated by automatic analysis tools. In the Mutation Miner system, we use the manually curated Swiss-Prot database to gain reliable grounding (see Section 4.2) of proteins found in biological documents.

Figure 2 shows the Swiss-Prot entry for a variant of the *xylanase 2* protein. The entries most important for NLP analysis are the various "Synonyms," as they can all appear in a given biomedical document, the canonical name ("Protein name") that can depend on its host organism, and a unique ID ("Primary accession number") that allows unambiguous linking to the protein's entry. The Swiss-Prot data can be downloaded from the Swiss-Prot website in XML, FASTA (Pearson and Lipman, 1988), and plain text format. The protein data is then encoded in the ontology, similar to the information concerning organisms. Thus, the ontology now has all the required information for detecting protein named entities, as well as assigning normalized names and grounding them to Swiss-Prot IDs.

Relation Instance Initialization. In order to support the detection, disambiguation, and confirmation of relations between organisms and proteins occurring in texts (discussed in Section 4.4), the ontology also needs instantiated relations between those entities. This can be achieved using another essential feature of Swiss-Prot, since its entries are cross-linked to other databases, notably to the NCBI Taxonomy database.

Using our previously created databases, we can infer the relations between proteins and organisms by means of the NCBI `TaxID` value, which is also transferred into our ontology. An example for this can be seen in the "From" line in Figure 2, where the ID of the host organism ("TaxID") is recorded. These relations form the basis for linking proteins found in documents to their hosting organisms (note the `organismProteinRel` relation in Figure 1).

Summary. At this stage, we have achieved an ontological data integration of two major biological databases, Swiss-Prot and the NCBI Taxonomy database. Furthermore, we manually added a number of instances relevant for the mutation domain as outlined in Section 3: amino acids, property descriptions, units of measurement, among others. In

[k] Swiss-Prot manual, `http://www.expasy.org/sprot/userman.html`

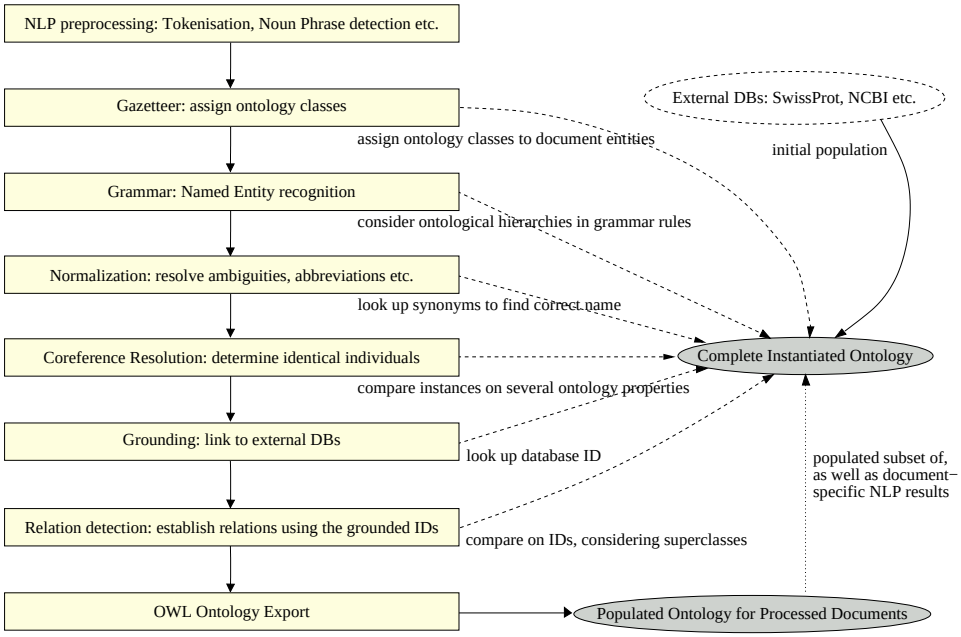

Figure 3 Workflow of the Mutation Miner NLP subsystem and its interaction with the ontology

this form, the ontology is already valuable for queries by a biologist or Semantic Web application. However, here we are concerned with the application of this knowledge source within a text mining application, in order to populate the ontology with instances that are *only* available in the biological literature. The resulting combination of curated knowledge with information described in natural language will thus create a new and unique knowledge base.

4 Text Mining and Ontology Population

This section discusses how to employ the modeled and initialized ontology for the various NLP analysis tasks within the Mutation Miner pipeline (see Figure 3). For the sake of brevity, we omit several standard NLP analysis steps in this discussion, like part-of-speech (POS) tagging, noun phrase (NP) chunking, or stemming. Readers unfamiliar with these tasks should consult (Hahn and Wermter, 2006) and the GATE user's guide.[1]

4.1 Named Entity Detection

The basic process in GATE for recognizing entities of a particular domain starts with the gazetteer component. It matches given lists of terms against the tokens of an analysed text and, in case of a match, adds an annotation named `Lookup` whose features depend on the list where the match was found. Its ontology-aware counterpart is the *OntoGazetteer*, which incorporates mappings between its term lists and ontology classes and assigns the proper

[1] GATE user's guide, `http://gate.ac.uk/sale/tao/index.html`

class in case of a term match. For example, using the instantiated Mutation Miner ontology, the gazetteer will annotate the text segment *Escherichia coli* with two `Lookup` annotations, having their `class` feature set to "Genus" for *Escherichia* and "Species" for *coli*.

In a second step, grammar rules written in the JAPE language are used to detect and annotate complex named entities. Those rules can refer to the `Lookup` annotation generated by the OntoGazetteer, and also evaluate the same ontology. For example, in a comparison like `class=="Species"`, the ontological hierarchy is taken into account so that also subspecies match, since a Subspecies *is-a* Species in the ontology. This can significantly reduce the overhead for grammar development and testing.

Similar processing takes place for detecting proteins, mutations, and other entities. The result of this stage is a set of named entities, which are, however, not yet normalized or grounded.

4.2 Normalization and Grounding

Normalization needs to decide on a canonical name for each entity, like a protein or an organism. Since the ontology encodes information about e.g. scientific names for organisms, a corresponding normalized entry can often be uniquely determined with a simple lookup. In case of abbreviations, however, finding the canonical name usually involves an additional disambiguation step.

For example, if we encounter *E. coli* in a text, it is first recognised as an organism from the pattern "species preceded by abbreviation." The NLP component can now query the ontology for a genus instance with a name matching `E*` and a species named `coli`, and filter the results for valid genus-species combinations denoting an existing organism. Ideally, this would yield the single combination of genus *Escherichia* and species *coli,* forming the correct organism name. However, the above query returns in fact four entries. Two can be discarded because their names are classified by NCBI as misspellings of *Escherichia coli*, as shown by the identical `tax_id` (cf. Table 2). Yet the two remaining combinations, with the names *Escherichia coli* and *Entamoeba coli,* are both classified as "scientific name." A disambiguation step now has to determine which one is the correct normalized form for *E. coli:* This is the task of coreference resolution covered in Section 4.3 below.

Once the normalized name (and thus the represented ontology instance) has been determined, in the case of organisms and proteins the corresponding database ID can be trivially retrieved from the instance, where it was stored as an OWL datatype property as described in Section 3.1. Since the database record can now be unambiguously looked up, the entity is grounded with respect to an external source. For our examples, these IDs are `P36217` for the xylanase variant shown in Figure 2, and `562` for *E. coli*, whose database entries are shown in Table 2.[m]

The end result of this step is a semantic annotation of the named entities as they appear in a text, which includes the detected information from normalization and grounding, as shown in Figure 4 for an Organism entity.

[m] Note that some additional processing is required for protein analysis, including abbreviation detection (Chang and Schütze, 2006), however, we cannot cover these steps within the scope of this paper.

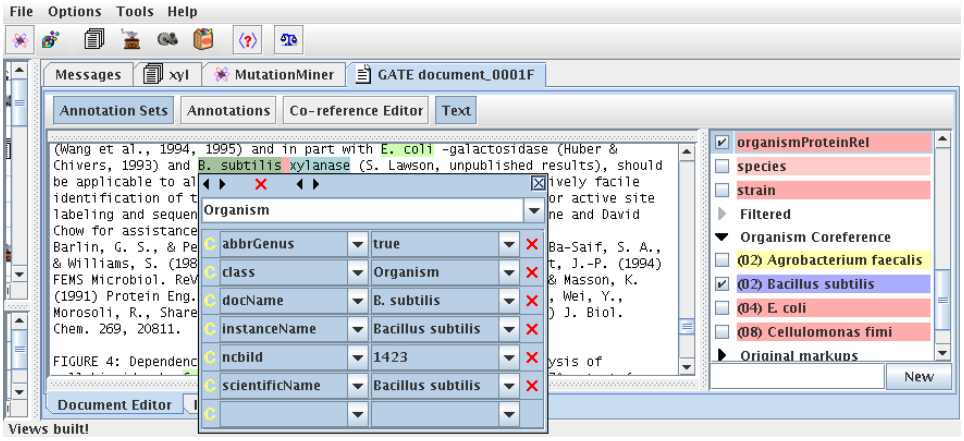

Figure 4 An organism annotation in GATE showing normalization and grounding of the textual entity *B. subtilis* to *Bacillus subtilis* with the NCBI database ID 1423

4.3 Coreference Resolution

Coreference resolution is another important step in a text mining system, as its results, coreference chains, form the basis for many downstream analysis tasks. Mutation Miner, for example, needs to identify the *impact* of a certain enzyme mutation. This requires the identification of *all* mentions of a mutation throughout the text, in order to examine their context, thereby extracting and

While coreference resolution has been studied extensively in the news domain (newspapers, newswires), the resolution of biological entities (both nominal and pronominal) is a rather new area of research. Here, we only focus on the ontological extensions of coreference resolution, not the basic approaches covered in the literature (Castaño et al., 2002; Gasperin, 2006; Kim and Park, 2004; Vlachos et al., 2006). In our system, we employ a fuzzy-based coreference resolution strategy using a number of heuristics that can use the instantiated ontology as a knowledge source. For example, coreference between an organism entity in abbreviated and several candidates in non-abbreviated form can be resolved by examining their context and picking the closest one of the candidates that was previously mentioned in non-abbreviated form. Entities that have been successfully grounded can be unambiguously identified as being equal by comparing their unique database IDs recorded in the ontology and thusly grouped in a coreference chain.

A common problem during coreference analysis are ambiguities occurring at the linguistic level. Here, the ontology can facilitate disambiguation by allowing comparisons considering different hierarchy levels in the ontology. For example, the NCBI Taxonomy database records the "parent" for each species. Thus, when testing for coreferring entities of an organism classified as "species" in the taxonomic tree, not only other species but also all subspecies can be taken into account by retrieving their parent IDs and using them in the comparison. For the subspecies *Batis mixta mixta*, for instance, the hierarchical relationship to its parent species *Batis mixta* can be established without resorting to substring tests by comparing the parent ID of the subspecies with the species' ID.

An example for successful coreference resolution on organisms can be seen in Figure 5,

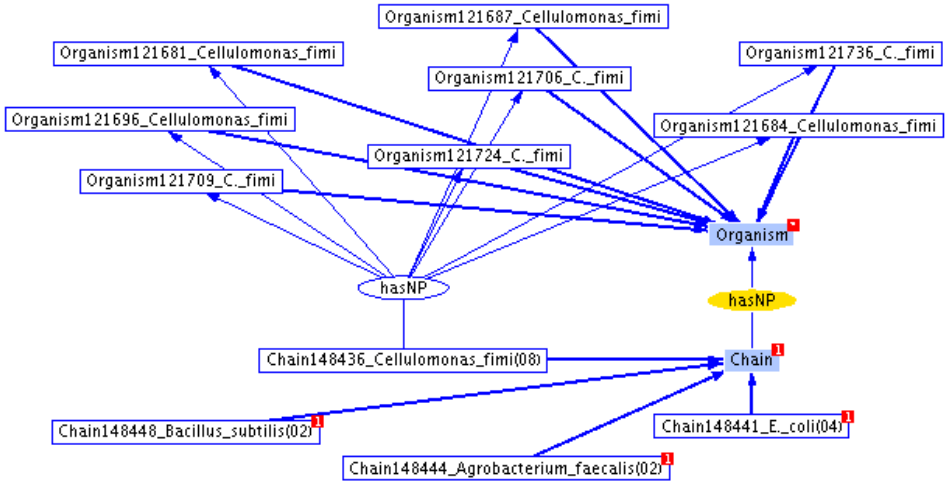

Figure 5 Organism coreference chains from the NLP-populated result ontology

which shows GrOWL[n] visualising a segment of the result ontology for a document, with ontology classes depicted by filled boxes and class instances by boxes with empty background. The *Chain* class representing coreference chains, here confined to Organism chains, is connected to its members by the object property `hasNP`. On the instance level, we see four chains, one for each organism found in the document. The chain for *Cellulomonas fimi* is expanded in the figure to show its eight members, which are instances of the Organism class. summarizing the impact descriptions.

4.4 Relation Detection

Relation detection, for example between organisms and proteins, requires more involved NLP analysis, like full or partial parsing for predicate-argument extraction (Hahn and Wermter, 2006; Leroy et al., 2003; Wattarujeekrit et al., 2004). A common problem in relation extraction is the high amount of ambiguity, especially when using full parsers (Yakushiji et al., 2001). Employing an ontology encoding semantically valid relations allows to constrain the number of detected relation candidates to the semantically valid ones, which ideally results in a unique relation and otherwise boosts precision (Leroy and Chen, 2005).

We give an example for detecting and disambiguating protein-organism relations, which is illustrated in Figure 6. Information from Swiss-Prot, including protein synonyms and taxonomic origin, is encoded in our ontology as detailed in Section 3.2. We can use this information to resolve ambiguous entities in a relation by discarding possible combinations that are not supported by the ontology, as each protein in Swiss-Prot is linked to its hosting organism via the latter's NCBI Taxonomy ID.

In the given example sentence, the phrase "*Bacillus subtilis* xylanase" refers to a protein of the xylanase family. This can be automatically determined by the named entity detection (see Section 4.1), semantically annotating "xylanase" as `Protein` and "*Bacillus subtilis*" as `Organism`. But it is not yet clear which protein is meant precisely. As can be seen in

[n] GrOWL ontology visualiser, `http://ecoinformatics.uvm.edu/dmaps/growl`

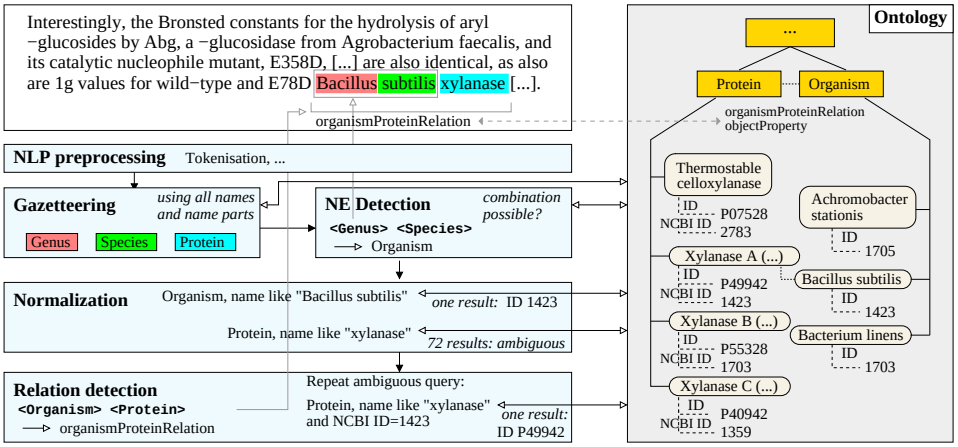

Figure 6 Protein disambiguation exploiting a detected relation

Figure 2, canonical protein names can change according to the organism they have been generated from: *xylanase 2* from *Trichoderma reesei* has the normalized name *Endo-1,4-beta-xylanase 2 [Precursor]* and a grounded ID in Swiss-Prot of P36217. Querying the ontology for proteins with "xylanase" in their name yields no less than 72 different proteins. However, in this example, *Bacillus subtilis,* which was tagged as organism by the NE component, can be unambiguously grounded, because it is a name occurring in the NCBI Taxonomy database, with the ID 1423 (see Figure 4).

So, the ontology query can be refined by including the organism's NCBI ID, which is used in Swiss-Prot to record the organism producing a protein. The resulting query for a protein named "∗xylanase∗" that is linked to the NCBI entry 1423 yields exactly one result, the correct protein *"Endo-1,4-beta-xylanase A precursor (EC 3.2.1.8) (Xylanase A) (1,4-beta-D-xylan xylanohydrolase A)."*

4.5 Exporting the Populated Ontology

Finally, the instances found in the document and the relations between them are exported to an OWL-DL ontology. Note that for the instances and relations available in the external databases, the result ontology is a subset of the one populated initially (cf. Figure 3).

In our implementation, ontology population is done by a custom GATE component, the *OwlExporter*, which is application domain-independent. It collects two special annotations, `OwlExportClass` and `OwlExportRelation`, which specify instances of classes and relations (i.e., object properties), respectively. These must in turn be created by application-specific components, since the decisions as to which annotations have to be exported, and what their OWL property values are, depend on the domain.

The class annotation carries the name of the class, a name for the instance like the Swiss-Prot official name for a protein, and the GATE internal ID of an annotation representing the instance in the document. If there are several occurrences of the same entity in the document, the final representation annotation is chosen from the ones in the coreference chain by the component creating the `OwlExportClass` annotation.

From the representative annotation, all further information is gathered. When it has

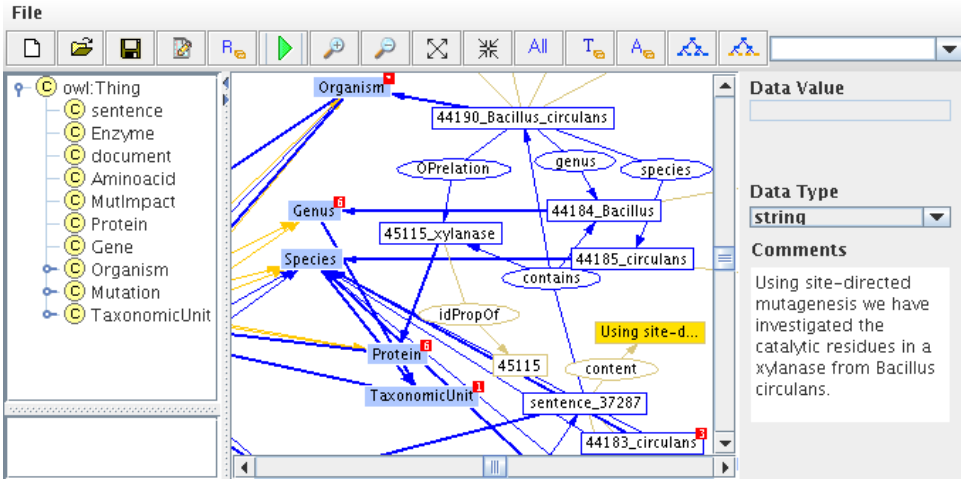

Figure 7 Mutation Miner ontology populated by NLP visualised in GrOWL

read the class name, OwlExporter queries the ontology via Jena for the properties of the class and then looks for equally named features in the representation annotation, using their values to set the OWL properties.

The exported, populated ontology also contains document-specific information; for example, for each class instance the sentence it was found in is recorded. Additional entity-specific information, like an automatically created summary for a mutation's impact, can also be exported. Figure 7 shows an excerpt of such an ontology populated by Mutation Miner, visualised using GrOWL.

5 Ontology Queries and Reasoning

In this section, we discuss the *access* to biological knowledge stored in the populated ontology, which now holds a combination of curated database information with instances created by targeted natural language analysis. Here, we focus mainly on scenarios relevant for an end user with a background in biology. However, many of the discussed access technologies also apply when the ontology becomes integrated in a larger Semantic Web context.

Within the scope of this paper, we cannot give a comprehensive introduction to ontology querying. We direct the reader to (Pan, 2007), which introduces the syntax and semantics of OWL and describes standard query languages like SPARQL and OWL-QL. Additionally, (Glimm and Horrocks, 2004) includes a comparison of several other queries languages, like DQL and nRQL.

5.1 Syntactic Queries using the RDF graph

OWL knowledge bases are encoded as RDF graphs, which can be queried using the SPARQL query language.[o] SPARQL employs SQL-like queries, e.g., for selecting instances based

[o] SPARQL RDF query language, `http://www.w3.org/TR/rdf-sparql-query/`

on their ID. For example, in order to construct a SPARQL query to the Mutation Miner ontology for the retrieval of the scientific name of the organism with NCBI ID 1423, one has to ask for a name (variable `?name`) that is the value of a `scientificName` property of an organism (variable `?organism`), which in turn also has an `ncbiTaxId` property with the value "1423":

```
SELECT ?name
WHERE { ?organism mm:scientificName ?name
        ?organism mm:ncbiTaxId 1423 }
```

More complex queries on RDF-graphs are of course also possible. However, SPARQL, by itself, is not OWL-capable in the sense that semantically richer queries considering the ontology classes and the class hierarchy cannot be expressed (e.g., formally restricting the queried subjects to instances of the Organism class is not possible). This kind of semantic queries involves A-Box (instances, individuals) reasoning and requires interfacing with a DL reasoner.

5.2 Semantic Queries using A-Box Reasoning

Semantic queries against an ontology are possible using description logic reasoners, such as *Racer* (Haarslev and Möller, 2001) or *Pellet*,[p] together with a query language like nRQL (Wessel and Möller, 2005) or RDQL.[q] In particular, these reasoners support the retrieval of individuals by taking the class hierarchy, object properties, and restrictions into account. OWL query languages are still in a state of flux, with new languages being proposed (like OWL-QL), thus the range of supported query languages differ for each DL reasoning system. In general, all these query languages use an SQL- or Lisp-like syntax, which makes them more agreeable to computer scientists than biologists. Recently, more user-friendly interface tools like *OntoIQ* (Baker et al., 2006b) have been developed that allow even users without knowledge of description logics to pose questions to an ontological knowledge base. Such functionality means that the scientific literature, processed by NLP tools to automatically populate ontology concepts, can subsequently be queried according to a user's familiarity with the ontology's domain.

An example for this is depicted in Figure 8, which shows the query interface of OntoIQ. The nRQL syntax of the query *"Find all sentences that describe mutations to the protein xylanase"* appears in the uppermost frame. The descriptors (Sentence, Mutation, and Protein) selected to appear in the query result are listed in the right hand frame below. The bottom frame shows the results returned through the interrogation of the NLP-populated Mutation Miner ontology that has been loaded into Racer. A user could now continue by examining the selected document sentences, connect with another ontology for further queries, or forward the selected instances to other (bioinformatics) tools for further automated processing.

Another example for accessing literature is shown in Figure 9, where the SWOOP[r] (Kalyanpur et al., 2005) ontology browser displays a selected instance from the organism class, hyperlinked with other information in the ontology. In particular, the example shows that a biologist can now access all information concerning an entity (here, the organism

[p] Pellet OWL-DL reasoner, `http://www.mindswap.org/2003/pellet/`

[q] RDQL Query Language for RDF, `http://www.w3.org/Submission/RDQL/`

[r] SWOOP Hypermedia OWL Editor/Browser, `http://www.mindswap.org/2004/SWOOP/`

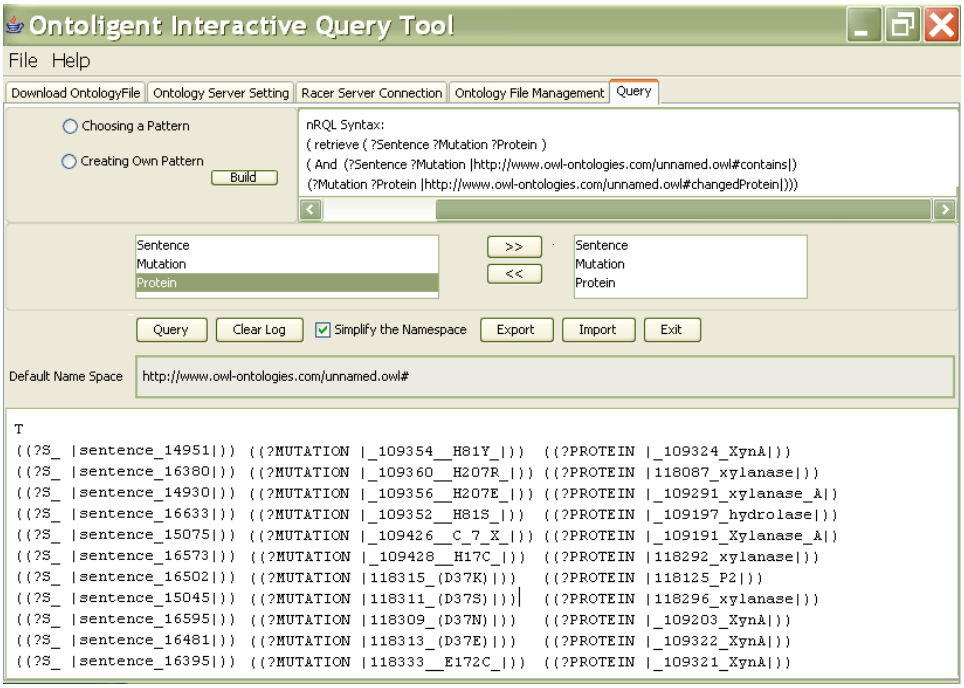

Figure 8 Querying an OWL-DL ontology populated by text mining full-text papers using the description logics reasoner *Racer* with the OntoIQ user interface

with the NCBI ID 5334), even when the exact name does not appear in a sentence, but only an abbreviation or other textual reference.

5.3 Natural Language Queries

Ontology query and navigation tools like OntoIQ and SWOOP are helpful for users that are familiar with the ontology and the domain, but not the formal query languages in use within the Semantic Web frameworks. To also support users that do not know the details of a specific ontology, it becomes desirable to allow questions to be posed in natural language to the ontology. Ontology-driven question-answering (QA) systems take an ontology and a question stated in a natural language as input and return answers based on the semantic data contained in the input ontology. Examples for such systems are Aqualog (Lopez et al., 2005) and ONLI (Kosseim et al., 2006). Although research in natural language interfaces to ontologies is still in its infancy, we believe that such systems have great potential for bringing in new users to the Semantic Web.

5.4 Querying Integrated Ontologies

The query paradigm becomes particularly interesting when the NLP-populated Mutation Miner ontology is integrated with other ontologies, as this allows cross-domain queries and reasoning. Instances generated by Mutation Miner alone provide information about impacts

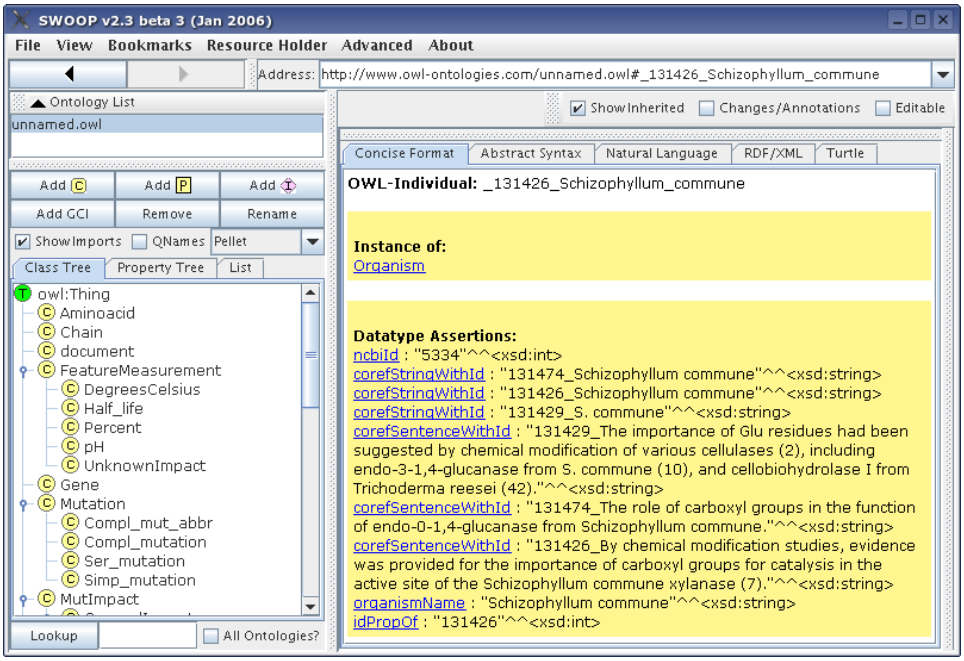

Figure 9 Browsing the populated ontology with SWOOP using the Pellet OWL-DL reasoner. Note the extracted sentences containing entities co-refering to the organism *Schizophyllum commune*.

of mutational change on protein performance. These instances permit queries such as: *"Find the locations of amino acids in xylanase proteins, which when mutated have resulted in enhanced enzyme thermostability."* Integration of the Mutation Miner ontology with the instantiated FungalWeb ontology (Shaban-Nejad et al., 2005) that represents knowledge about the enzyme industry and fungal species additionally permits cross-disciplinary queries. For example, queries asking *"Identify the industrial benefits derived from commercial enzyme products based on mutated xylanases"* or *"What commercial enzyme products are not the result of mutational improvement"* become now possible. Depending on the user, access to this knowledge can assist in decision making for experimental design or product development. We also envision that other ontologies, such as the instantiated Protein Ontology[s] could be integrated with the Mutation Miner ontology, which would enable a wider range of protein-specific A-box queries. For further examples illustrating the use of formal ontology reasoning and querying in concrete application scenarios from fungal biotechnology, we refer the reader to (Baker et al., 2005, 2006a).

6 Evaluation

In this section, we present first results of the performance of our approach. So far, we evaluated our ontological text mining system on two corpora: (i) a primary set of 20 documents describing mutations on the *xylanase* protein family, and (ii) a supplementary set of 6 documents describing mutations on the *haloalkane dehalogenase* protein family. A

[s]Protein Ontology, `http://www.proteinontology.info/`

detailed discussion and comparison with other systems is presented in the next section.

Initialized Ontology. The initialized Mutation Miner ontology (cf. Section 3.2) contains 1,350,000 RDF triples. Table 3 shows an overview of the main classes with the number of generated instances. Note that strains are only partially recognized through ontological strain information, we employ additional JAPE grammar rules to detect a variety of strain formats. Interestingly, although a large number of proteins are described on Swiss-Prot, they reference only a comparatively small number of organisms in the NCBI Taxonomy database (O/P relations).

Table 3 Size of the initialized Mutation Miner ontology

Ontology class	**Organism**	**Genus**	**Species**	**Strain**	**Protein**	**O/P Relation**
Number of instances	310,774	41,475	131,765	1053	222,289	9878

NLP-Instantiated Ontology. We now give qualitative results of the NLP performance for the *xylanase* protein family. Table 4 shows how many entities were detected correctly, as well as precision, recall, and F-measure,[t] for each of the main classes *organism, protein,* and *mutation*, compared with the manually annotated gold standard. These results were obtained using GATE's *corpus benchmark tool*.

Table 4 Performance of the ontological named entity recognition on the *xylanase* corpus

Class	**Correct**	**Partial**	**Missing**	**Spurious**	**Precision**	**Recall**	**F-Measure**
Organism	414	94	41	5	0.90	0.84	0.87
Protein	942	240	708	921	0.51	0.56	0.53
Mutation	648	71	143	46	0.89	0.79	0.84

Grounding Performance. We now investigate how well our system performs the grounding and normalization tasks, i.e., finding an external database ID for each entity (NCBI for organisms, Swiss-Prot for proteins). The results for organisms are given in Table 5. Here, *accuracy* is defined as the ratio of correctly normalized entities over correctly detected entities.

Table 5 Normalization and grounding performance for organisms on the *xylanase* corpus

Class	**Precision**	**Recall**	**F-Measure**	**Accuracy**
Organism	0.85	0.80	0.82	0.95

Evaluating the protein performance is somewhat more complex, as this involves for each detected protein entity: (i) finding its host organism (protein/organism relation detection), (ii) normalizing the protein with respect to the detected relation, and (iii) grounding it in Swiss-Prot. Only if all three sub-tasks were successful, we count a positive hit for a protein (i.e., we do not evaluate each step by itself). Additionally, for the Mutation Miner scenario, we distinguish three classes of proteins discussed in the protein engineering literature: type A proteins, which have been the subject of mutational experiments within a paper; type B proteins, which are used in a paper for comparisons to the mutated proteins; and type C proteins, which are all other proteins mentioned in a paper (e.g., appearing in the *discussion* or *references* sections). Obviously, our focus is on obtaining a high recall of class A proteins, which is a different objective from a general protein recognition experiment. Table 6 shows the results for the *xylanase* and *haloalkane dehalogenase* corpora.

[t] Here, F-measure = $2 * p * r/(p+r)$

Table 6 Combined relation detection, normalization, and grounding performance for proteins on the *xylanase* and *haloalkane dehalogenase* corpora

Corpus	Type	Precision	Recall	F-Measure
xylanase	A	0.15	0.59	0.24
	A + B	0.32	0.57	0.41
	A + B + C	0.43	0.33	0.37
haloalkane dehalogenase	A	0.42	0.89	0.57
	A + B	0.42	0.78	0.55
	A + B + C	0.51	0.75	0.61

Note that we did not perform different NLP analyses in this evaluation, rather all detected proteins are included for each sub-type, hence the lower precision for type A and A+B.

7 Related Work and Discussion

There exists some previous work concerning the detection of mutations in texts, however, the results are not directly comparable with our system. MuteXt (Horn et al., 2004) focuses on single-point mutations for G protein-coupled receptors and nuclear hormone receptors, whereas we are concerned with single- and multiple-point mutations of enzymes. Their reported results only include the mutation extraction performance. Likewise, the system described by (Rebholz-Schuhmann et al., 2004) focuses only on the extraction of mutation-gene pairs. None of the existing systems are concerned with ontology-based processing of texts or ontology population for further querying of text segments using DL reasoners.

Named entity normalization was one of the tasks carried out within the *Critical Assessment for Information Extraction systems in Biology* (BioCreAtIvE)[u] 2004 competition. However, there was no task for the normalization of organism or protein names. In general, a performance of 70–90% F-measure was reported on gene name normalization (Hirschman et al., 2005). Preliminary findings on the normalization of human proteins report 50–60% F-measure (Hirschmann, 2005). Note that our results additionally include the *grounding* of organism and protein names, which is an additional task not evaluated within BioCreAtIvE.

Another approach that also grounds proteins to Swiss-Prot entries is BioAR (Kim and Park, 2004). However, in addition to the nominal coreference resolution performed by our system, the authors additionally integrated pronominal resolution. They report a 59% precision and 40.7% recall on protein grounding in Swiss-Prot (Kim and Park, 2004), which is roughly comparable to our results (which have generally higher recall at the expense of precision).

A number of approaches combine text mining with existing ontologies, like the *Gene Ontology* (GO), to annotate database entries with segments from the literature (Camon et al., 2005; Couto et al., 2003; Stoica and Hearst, 2006). Although our approach uses existing databases, our approach goes beyond these systems in that we also extract and annotate information not available in databases, which could be used to curate new entries, like for the PMD in our application scenario.

It is also important to distinguish our approach from the field of ontology learning, where NLP is used to determine potential classes and their relations from texts (Buitelaar et al., 2005; Staab and Studer, 2004). Presently, we do not consider these technologies advanced enough to deal with the stated requirements of precise biological entity detection,

[u] BioCreAtIvE, `http://biocreative.sourceforge.net/`

normalization, and grounding. However, a hybrid approach might be able to dynamically adapt to new research topics while still delivering the advantages of linking extracted information with databases.

Ontological information retrieval systems apply NLP to automatically link documents to existing ontologies. Examples for this category are systems like Textpresso (Müller et al., 2004) and GoPubMed (Doms and Schroeder, 2005). It is important to note that while our approach can also retrieve documents based on queries, it delivers much more fine-grained results, down to individual entities detected in sentences. Hence, we provide for additional capabilities like entity extraction, which could in the future be enhanced to include automatic summarization of biological documents based on a (natural language) query, similar to the tasks of the NIST-sponsored Document Understanding Conference (DUC).[v] See (Witte et al., 2006) for our previous work in this area.

Contemporary approaches to the interoperability of biological content are rooted in relational database technology and its query languages. Public biological information resources are typically accessible through web-based forms, which employ SQL-queries against a single or multiple databases. More recently, a trend towards online query access has emerged, since some biological databases now provide remote access to their content in standardized formats like XML, which can be queried by languages like XQuery[w] or Xcerpt.[x] An even smaller number of biological databases, notably Swiss-Prot, YeastHub, and Linkhub have content available in semantically richer knowledge representation formats like RDF. However, the majority (an estimated 80%) of scientific information is not available in curated biological databases, but only in unstructured, natural language form. This is particularly the case for information related to protein mutations and their impacts, despite the existence of customized databases with manually derived content, such as BRENDA (Schomburg et al., 2004), the Protein Mutant Database (PMD), and OMIM. Our approach seeks to bypass both manual curation and low-precision Information Retrieval (IR) of scientific literature, by employing advanced text mining technologies for information extraction. By coupling the results with standardized ontology formats for high granularity knowledge representation, accessible with equally expressive, off-the-shelf tools that provide user-friendly query interfaces, we significantly enhanced the knowledge discovery process for biologists.

8 Conclusions and Future Work

This paper describes the combination of two still emerging technologies, Semantic Web Ontologies and Text Mining. An important dimension to the proliferation of the Semantic Web in the life science domain is the availability of scientific content to Semantic Web tools. Document-centric access to the literature is still the predominant form of user access. In this paper, we have illustrated how to make database content and excerpts of the scientific literature available for the Semantic Web and described the different query paradigms that are available for scientists accessing this information. The feasibility of this approach, implemented with state-of-the-art tools, has been demonstrated for a relevant application scenario, protein engineering literature. We consider this approach an important advancement that will indelibly impact upon the uptake of the Semantic Web, particularly

[v] Document Understanding Conference (DUC), `http://duc.nist.gov/`
[w] W3C XML Query (XQuery), `http://www.w3.org/XML/Query/`
[x] Xcerpt rule-based query language, `http://www.xcerpt.org/`

by life scientists.

While several improvements are possible to our system, most notably to the protein recognition performance, these are details that do not obstruct our main goal, the query paradigm for accessing biological knowledge through NLP-driven OWL-DL ontology population. This is particularly important as description logic (DL)-based queries and their results can also be interpreted as solutions to biological application problems, as shown in (Wolstencroft et al., 2007). We expect that DL-based queries and reasoning will in the future become integral parts of text mining systems. Thus, we do not agree with the viewpoints presented in (Tsujii and Ananiadou, 2005), which considers thesauri, and not ontologies, to be sufficient for (biological) text mining.

The emergence of ontological NLP is also likely to give rise to an increase in the abundance of instantiated ontologies serving as knowledge bases. Given that the scientific community can see beyond the challenges of new query tools and workflows for information retrieval, it is reasonable to expect that NLP techniques connected with ontologies will contribute significantly to the discovery processes in the life sciences.

Acknowledgements

The authors would like to thank Vladislav Ryzhikov for his contributions to the Mutation Miner NLP subsystem. Jirí Damborský is acknowledged for sharing his expert knowledge of the haloalkane dehalogenase literature.

References

Ananiadou, S. and McNaught, J. (Eds.) (2006) *Text Mining for Biology and Biomedicine*, Artech House.

Bairoch, A., Apweiler, R., Wu, C.H., Barker, W.C., Boeckmann, B., Ferro, S., Gasteiger, E., Huang, H., Lopez, R., Magrane, M., Martin, M.J., Natale, D.A., O'Donovan, C., Redaschi, N. and Yeh, L.S.L. (2005) 'The Universal Protein Resource (UniProt)', *Nucleic Acids Research*.

Baker, C.J.O. and Cheung, K.H. (Eds.) (2007) *Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences*, Springer Science+Business Media, New York, NY, USA.

Baker, C.J.O., Shaban-Nejad, A., Su, X., Haarslev, V. and Butler, G. (2006a) 'Semantic Web Infrastructure for Fungal Enzyme Biotechnologists', *Journal of Web Semantics*, Vol. 4, No. 3. Special issue on Semantic Web for the Life Sciences.

Baker, C.J.O., Su, X., Butler, G. and Haarslev, V. (2006b) 'Ontoligent Interactive Query Tool', In M.T. Koné and D. Lemire (Eds.) *Canadian Semantic Web Series*, Springer, *Semantic Web and Beyond*, Vol. 2.

Baker, C.J.O. and Witte, R. (2006) 'Mutation Mining—A Prospector's Tale', *Information Systems Frontiers (ISF)*, Vol. 8, No. 1, pp. 47–57.

Baker, C.J.O., Witte, R., Shaban-Nejad, A., Butler, G. and Haarslev, V. (2005) 'The FungalWeb Ontology: Application Scenarios', In *Eighth Annual Bio-Ontologies Meeting*, Detroit, Michigan, USA, pp. 1–2.

Bodenreider, O. (2006) 'Lexical, Terminological, and Ontological Resources for Biological Text Mining', In (Ananiadou and McNaught, 2006), Chap. 3, pp. 43–66.

Bontcheva, K., Tablan, V., Maynard, D. and Cunningham, H. (2004) 'Evolving GATE to Meet New Challenges in Language Engineering', *Natural Language Engineering*, Vol. 4, No. 3/4, pp. 349–373.

Buitelaar, P., Cimiano, P. and Magnini, B. (Eds.) (2005) *Ontology Learning from Text: Methods,*

Evaluation and Applications, *Frontiers in Artificial Intelligence and Applications*, Vol. 123, IOS Press.

Camon, E.B., Barrell, D.G., Dimmer, E.C., Lee, V., Magrane, M., Maslen, J., Binns, D. and Apweiler, R. (2005) 'An evaluation of GO annotation retrieval for BioCreAtIvE and GOA', *BMC Bioinformatics*, Vol. 6, No. Suppl 1:S17.

Castaño, J., Zhang, J. and Pustejovsky, J. (2002) 'Anaphora Resolution in Biomedical Literature', In *International Symposium on Reference Resolution*, Alicante, Spain.

Chang, J. and Schütze, H. (2006) 'Abbreviations in Biomedical Text', In (Ananiadou and McNaught, 2006), Chap. 5.

Cohen, A.M. and Hersh, W.R. (2005) 'A survey of current work in biomedical text mining', *Briefings in Bioinformatics*, Vol. 6, pp. 57–71.

Couto, F.M., Silva, M.J. and Coutinho, P. (2003) 'ProFAL: PROtein Functional Annotation through Literature', In *VII Conference on Software Engineering and Databases (JISBD)*, pp. 747–756.

Cunningham, H., Maynard, D., Bontcheva, K. and Tablan, V. (2002) 'GATE: A framework and graphical development environment for robust NLP tools and applications', In *Proceedings of the 40th Anniversary Meeting of the ACL*. `http://gate.ac.uk`.

Cunningham, H., Maynard, D. and Tablan, V. (2000) 'JAPE: a Java Annotation Patterns Engine (Second Edition)', Tech. rep., University of Sheffield, Department of Computer Science.

Doms, A. and Schroeder, M. (2005) 'GoPubMed: Exploring PubMed with the GeneOntology', *Nucleic Acids Research*, Vol. 33, pp. W783–W786.

Federhen, S. (2003) 'The Taxonomy Project', In J. McEntyre and J. Ostell (Eds.) *The NCBI Handbook*, National Library of Medicine (US), National Center for Biotechnology Information, Chap. 4.

Gabdoulline, R.R., Ulbrich, S., Richter, S. and Wade, R.C. (2006) 'ProSAT2–Protein Structure Annotation Server', *Nucleic Acids Research*, Vol. 34, No. suppl_2, pp. W79–W83.

Gasperin, C. (2006) 'Semi-supervised anaphora resolution in biomedical texts', In *Proceedings of the HLT-NAACL Workshop on Linking Natural Language Processing and Biology (BioNLP)*, New York City, NY, USA.

Glimm, B. and Horrocks, I. (2004) 'Query answering systems in the semantic web', In *KI-2004 Workshop on Applications of Description Logics (ADL)*, Ulm, Germany.

Haarslev, V. and Möller, R. (2001) 'RACER System Description', In *Proceedings of International Joint Conference on Automated Reasoning (IJCAR)*, Springer-Verlag Berlin, Siena, Italy, pp. 701–705.

Hahn, U. and Wermter, J. (2006) 'Levels of Natural Language Processing for Text Mining', In (Ananiadou and McNaught, 2006), Chap. 2, pp. 13–41.

Hirschman, L., Yeh, A., Blaschke, C. and Valencia, A. (2005) 'Overview of BioCreAtIvE: critical assessment of information extraction for biology', *BMC Bioinformatics*, Vol. 6, No. Suppl 1.

Hirschmann, L. (2005) 'Mapping from Text to Ontology for Biological Applications', Talk at Knowledge-Based Bioinformatics Workshop (KBB), Montréal, Canada.

Horn, F., Lau, A.L. and Cohen, F.E. (2004) 'Automated extraction of mutation data from the literature: application of MuteXt to G protein-coupled receptors and nuclear hormone receptors', *Bioinformatics*, Vol. 20, No. 4, pp. 557–568.

Kalyanpur, A., Parsia, B., Sirin, E., Cuenca-Grau, B. and Hendler, J. (2005) 'Swoop: A 'Web' Ontology Editing Browser', *Journal of Web Semantics*, Vol. 4, No. 2.

Kawabata, T., Ota, M. and Nishikawa, K. (1999) 'The protein mutant database', *Nucleic Acids Research*, Vol. 27, No. 1.

Kim, J.J. and Park, J.C. (2004) 'BioAR: Anaphora Resolution for Relating Protein Names to Proteome Database Entries', In S. Harabagiu and D. Farwell (Eds.) *ACL 2004: Workshop on Reference Resolution and its Applications*, Association for Computational Linguistics, Barcelona, Spain, pp. 79–86.

Kosseim, L., Siblini, R., Baker, C.J.O. and Bergler, S. (2006) 'Using Selectional Restrictions to Query an OWL Ontology', In *International Conference on Formal Ontology in Information Systems (FOIS 2006)*, Baltimore, Maryland, USA.

Lambrix, P., Tan, H., Jakoniene, V. and Strömbäck, L. (2007) 'Biological Ontologies', In (Baker and Cheung, 2007), Chap. 4, pp. 85–99.

Leroy, G. and Chen, H. (2005) 'Genescene: An Ontology-enhanced Integration of Linguistic and Co-occurrence based Relations in Biomedical Texts', *Journal of the American Society for Information Systems and Technology (JASIST)*, Vol. 56, No. 5, pp. 457–468.

Leroy, G., Chen, H. and Martinez, J.D. (2003) 'A shallow parser based on closed-class words to capture relations in biomedical text', *J. of Biomedical Informatics*, Vol. 36, pp. 145–158.

Lopez, V., Pasin, M. and Motta, E. (2005) 'AquaLog: An Ontology-Portable Question Answering System for the Semantic Web', In *Proceedings of the European Semantic Web Conference (ESWC)*, Crete, pp. 546–562.

Müller, H.M., Kenny, E.E. and Sternberg, P.W. (2004) 'Textpresso: An Ontology-Based Information Retrieval and Extraction System for Biological Literature', *PLoS Biology*, Vol. 2, No. 11, pp. 1984–1998.

Obrst, L., Ceusters, W., Mani, I., Ray, S. and Smith, B. (2007) 'The Evaluation of Ontologies: Toward Improved Semantic Interoperability', In (Baker and Cheung, 2007), pp. 139–158.

Pan, J.Z. (2007) 'OWL for the novice: A logical perspective', In (Baker and Cheung, 2007), Chap. 8, pp. 159–182.

Pearson, W.R. and Lipman, D.J. (1988) 'Improved tools for biological sequence comparison', *Proceedings of the National Academy of Sciences of the USA*, Vol. 85, No. 8, pp. 2444–2448.

Rebholz-Schuhmann, D., Kirsch, H. and Couto, F. (2005) 'Facts from Text—Is Text Mining Ready to Deliver?', *PLoS Biology*, Vol. 3, pp. 188–191.

Rebholz-Schuhmann, D., Marcel, S., Albert, S., Tolle, R., Casari, G. and Kirsch, H. (2004) 'Automatic extraction of mutations from Medline and cross-validation with OMIM', *Nucleic Acids Research*, Vol. 32, No. 1, pp. 135–142.

Schomburg, I., Chang, A., Ebeling, C., Gremse, M., Heldt, C., Huhn, G. and Schomburg, D. (2004) 'BRENDA, the enzyme database: updates and major new developments', *Nucleic Acids Research*, Vol. 32.

Shaban-Nejad, A., Baker, C.J.O., Haarslev, V. and Butler, G. (2005) 'The FungalWeb Ontology: Semantic Web Challenges in Bioinformatics and Genomics', In *Springer LNCS 3729*, pp. 1063–1066.

Smith, M.K., Welty, C. and McGuinness, D.L. (Eds.) (2004) *OWL Web Ontology Language Guide*, World Wide Web Consortium. `http://www.w3.org/TR/owl-guide/`.

Staab, S. and Studer, R. (Eds.) (2004) *Handbook on Ontologies*, Springer.

Stoica, E. and Hearst, M. (2006) 'Predicting Gene Functions from Text Using a Cross-Species Approach', In *Pacific Symposium on Biocomputing (PSB)*, pp. 88–99.

Tsujii, J. and Ananiadou, S. (2005) 'Thesaurus or logical ontology, which one do we need for text mining?', *Language Resources and Evaluation*, Vol. 39, No. 1, pp. 77–90.

Vlachos, A., Gasperin, C., Lewin, I. and Briscoe, T. (2006) 'Bootstrapping the Recognition and Anaphoric Linking of Named Entities in Drosophila Articles', In *Pacific Symposium on Biocomputing*, pp. 100–111.

Wattarujeekrit, T., Shah, P.K. and Collier, N. (2004) 'PASBio: predicate-argument structures for event extraction in molecular biology', *BioMed Central Bioinformatics*, Vol. 5, No. 155.

Wessel, M. and Möller, R. (2005) 'High Performance Semantic Web Query Answering Engine', In *International Workshop on Description Logics (DL)*, Edinburgh, Scotland, UK.

Witte, R. and Baker, C.J.O. (2005) 'Combining Biological Databases and Text Mining to support New Bioinformatics Applications', In *10th International Conference on Applications of Natural Language to Information Systems (NLDB)*, Springer, *LNCS*, Vol. 3513, pp. 310–321.

Witte, R., Kappler, T. and Baker, C.J.O. (2007) 'Ontology Design for Biomedical Text Mining', In (Baker and Cheung, 2007), Chap. 13, pp. 281–313.

Witte, R., Krestel, R. and Bergler, S. (2006) 'Context-based Multi-Document Summarization using Fuzzy Coreference Cluster Graphs', In *Proceedings of Document Understanding Work-*

shop (DUC), New York City, NY, USA. `http://www-nlpir.nist.gov/projects/duc/pubs/2006papers/20.final.pdf`.

Wolstencroft, K., Stevens, R. and Haarslev, V. (2007) 'Applying OWL Reasoning to Genomic Data', In (Baker and Cheung, 2007), Chap. 11, pp. 225–248.

Yakushiji, A., Tateisi, Y., Miyao, Y. and Tsujii, J. (2001) 'Event extraction from biomedical papers using a full parser', In *Proceedings of the 6th Pacific Symposium on BioComputing (PSB)*, Hawaii, USA, pp. 408–419.