Annotation and Disambiguation of Semantic Types in Biomedical Text: a Cascaded Approach to Named Entity Recognition

Dietrich Rebholz-Schuhmann, Harald Kirsch, Sylvain Gaudan, Miguel Arregui
European Bioinformatics Institute (EBI),
Wellcome Trust Genome Campus, Hinxton, Cambridge, UK
{rebholz,kirsch,gaudan,arregui}@ebi.ac.uk

Goran Nenadic
School of Informatics
University of Manchester
Manchester, UK
g.nenadic@manchester.ac.uk

Abstract

Publishers of biomedical journals increasingly use XML as the underlying document format. We present a modular text-processing pipeline that inserts XML markup into such documents in every processing step, leading to multi-dimensional markup. The markup introduced is used to identify and disambiguate named entities of several semantic types (protein/gene, Gene Ontology terms, drugs and species) and to communicate data from one module to the next. Each module independently adds, changes or removes markup, which allows for modularization and a flexible setup of the processing pipeline. We also describe how the cascaded approach is embedded in a large-scale XML-based application (EBIMed) used for on-line access to biomedical literature. We discuss the lessons learnt so far, as well as the open problems that need to be resolved. In particular, we argue that the pragmatic and tailored solutions allow for reduction in the need for overlapping annotations — although not completely without cost.

1 Introduction

Publishers of biomedical journals have widely adopted XML as the underlying format from which other formats, such as PDF and HTML, are generated. For example, documents in XML format are available from the National Library of Medicine[1] (Medline abstracts and Pubmed[2] Central documents), and from BioMed Central[3] (full text journal articles). Other publishers are heading into the same direction. Such documents contain logical markup to organize meta-information such as title, author(s), sections, headings, citations, references, etc. Inside the text of a document, XML is used for physical markup, e.g. text in italic or boldface, subscript and superscript insertions, etc. Manually generated semantic markup is available only on the document level (e.g. MeSH terms).

One of the most distinguished feature of scientific biomedical literature is that it contains a large amount of terms and entities, the majority of which are explained in public electronic databases. Terms (such as names of genes, proteins, gene products, organisms, drugs, chemical compounds, etc.) are a key factor for accessing and integrating the information stored in literature (Krauthammer and Nenadic, 2004). Identification and markup of names and terms in text serves several purposes:

(1) The users profit from highlighted semantic types, e.g. protein/gene, drug, species, and from links to the defining database for immediate access and exploration.

(2) Identified terms facilitate and improve statistical and NLP based text analysis (Hirschman et al., 2005; Kirsch et al., 2005).

In this paper we describe a cascaded approach to named-entity recognition (NER) and markup in biomedicine that is embedded into EBIMed[4], an on-line service to access the literature (Rebholz-Schuhmann et al., forthcoming). EBIMed facilitates both purposes mentioned above. It keeps the annotations provided by publishers and inserts XML annotations while processing the text. Named entities from different resources are identified in the text. The individual modules provide annotation of protein names with unique identifiers, disambiguation of protein names that are ambiguous acronyms, annotation of drugs, Gene Ontology[5] terms and species. The identification of protein named entities can be further used in an alternative pipeline to identify events

[1] National Library of Medicine, http://www.nlm.nih.gov/
[2] PubMed, http://www.pubmed.org
[3] BioMed Central Ltd, http://www.biomedcentral.com/
[4] EBIMed, www.ebi.ac.uk/Rebholz-srv/ebimed
[5] GO, Gene Ontology, http://geneontology.org, (GO consortium, 2005).

such as protein-protein interactions and associations between terms and mutations (Blaschke et al., 1999; Rzhetsky et al., 2004; Rebholz-Schuhmann et al., 2004; Nenadic and Ananiadou, 2006).

The rest of the paper is organised as follows. In Section 2 we briefly discuss problems with biomedical NER. The cascaded approach and an online text mining system are described in sections 3 and 4 respectively. We discuss the lessons learnt from the on-line application and remainig open problems in Section 5, while conclusions are presented in Section 6.

2 Biomedical Named Entity Recognition

Terms and named-entities (NEs) are the means of scientific communication as they are used to identify the main concepts in a domain. The identification of terminology in the biomedical literature is one of the most challenging research topics both in the NLP and biomedical communities (Hirschman et al., 2005; Kirsch et al., 2005).

Identification of named entities (NEs) in a document can be viewed as a three-step procedure (Krauthammer and Nenadic, 2004). In the first step, single or multiple adjacent words that indicate the presence of domain concepts are recognised (*term recognition*). In the second step, called *term categorisation*, the recognised terms are classified into broader domain classes (e.g. as genes, proteins, species). The final step is *mapping* of terms into referential databases. The first two steps are commonly referred to as *named entity recognition (NER)*.

One of the main challenges in NER is a huge number of new terms and entities that appear in the biomedical domain. Further, terminological variation, recognition of boundaries of multiword terms, identification of nested terms and ambiguity of terms are the difficult issues when mapping terms from the literature to biomedical database entries (Hirschman et al., 2005; Krauthammer and Nenadic, 2004).

On one hand, NER in the biomedical domain (in particular the recognition part) profits from large, freely available terminological resources, which are either provided as ontologies (e.g. Gene Ontology, ChEBI[6], UMLS[7]) or result from biomedical databases containing named entities (e.g. UniProt/Swiss-Prot[8]). On the other hand, combining sets of terms from different terminological resources leads to naming conflicts such as homonymous use of names and terminological ambiguities. The most obvious problem is when the same span of text is assigned to different semantic types (e.g. '*rat*' denotes a species and a protein). In this case, there are three types of ambiguities:

(**Amb1**) A name is used for different entries in the same database, e.g. the same protein name serves for a given protein in different species (Chen et al., 2005).

(**Amb2**) A name is used for entries in multiple databases and thus represents different types, e.g. '*rat*' is a protein and a species.

(**Amb3**) A name is not only used as a biomedical term but also as part of common English (in contrast to the biomedical terminology), e.g. '*who*' and '*how*', which are used as protein names.

In some cases (i.e. Amb2), broader classification can help to disambiguate between different entries (e.g. differentiate between '*CAT*' as a protein, animal or medical device). However, it is ineffective in situations where names can be mapped to several different entries in the same data source. In such situations, disambiguation on the resource level is needed (see, for example, (Liu et al., 2002) for disambiguation of terms associated with several entries in the UMLS Metathesaurus).

In many solutions, the three steps in biomedical NER (namely, recognition, categorisation and mapping to databases) are merged within one module. For example, using an existing terminological database for recognition of NEs, effectively leads to complete term identification (in cases where there are no ambiguities). Some researchers, however, have stressed the advantages of tackling each step as a separate task, pointing at different sources and methods needed to accomplish each of the subtasks (Torii et al., 2003; Lee et al., 2003). Also, in the case of modularisation, it is easier to integrate different solutions for each specific problem. However, it has been suggested that whether a clear separation into single steps would improve term identification is an open issue (Krauthammer and Nenadic, 2004). In this paper we discuss a cascaded, modular approach to biomedical NER.

3 Biomedical NER based on XML annotation: Modules in a pipeline

In this Section we present a modular approach to identification, disambiguation and annotation of

[6] ChEBI, Chemical Entities of Biological Interest, http://www.ebi.ac.uk/chebi/m

[7] UMLS, Unified Medical Language System http://www.nlm.nih.gov/research/umls/, (Browne et al., 2003).

[8] UniProt, http://www.ebi.uniprot.org/, (Bairoch et al., 2005); Swiss-Prot, http://ca.expasy.org/sprot/

several biomedical semantic types in the text. Full identification of NEs and resolving ambiguities in particular, may require a full parse tree of a sentence in addition to the analysis of local context information. On the other hand, full parse trees may be only derivable after NEs are resolved. Methods to efficiently overcome these problems are not yet available today and in order to come up with an applicable solution, it was necessary to choose a more pragmatic approach.

We first discuss the basic principles and design of the processing pipeline, which is based on a pragmatic cascade of modules, and then present each of the modules separately.

3.1 Modular design of a text processing pipeline

Our methodology is based on the idea of separating the process into clearly defined functions applied one after another to text, in a processing *pipeline* characterized by the following statements:

(**P1**) The complete text processing task consists of separate and independent modules.

(**P2**) The task is performed by running all modules exactly once in a fixed sequence.

(**P3**) Each module operates continuously on an input *stream* and performs its function on stretches or "windows" of text that are usually much smaller than the whole input. As soon as a window is processed, the module produces the resulting output.

(**P4**) After the startup phase, all modules run in parallel. Incoming requests for annotation are accepted by a master process that ensures that all required modules are approached in the right order.

(**P5**) Communication of information between the modules is strictly downstream and all meta-information is contained in the data stream itself in the form of XML markup.

An instance of a processing pipeline (which is actually embedded in EBIMed) is presented in Figure 1. The modules M-1 to M-8 are run in this order, and no communication between them is needed apart from streaming the text from the output of one module to the input of another. The text contains the meta-data as XML markup. The modules are described below.

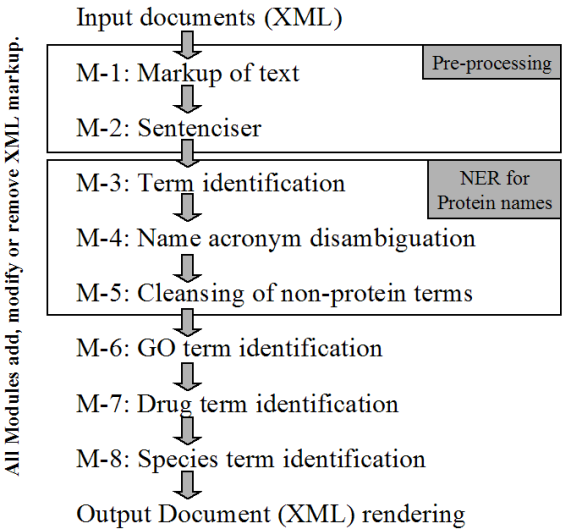

Figure 1. A processing pipeline embedded in EBIMed

Although this is the standard pipeline for EBIMed, it is possible to re-arrange the modules to favour identification of specific semantic types. More precisely, in our modular approach, after identification of a term in the text, disambiguation only decides whether the term is of that type or not. If it is not, the specific annotation is removed and left to the downstream modules to tag the term differently. While this requires *n* identification steps, adding identification of new types is independent of modules already present. However, the prioritization of semantic types is enforced by the order of the associated term identification modules.

3.2 Input documents and pre-processing

Input documents are XML-formatted Medline abstracts as provided from the National Library of Medicine (NLM). The XML structure of Medline abstracts includes meta information attached to the original document, such as the journal, author list, affiliations, publication dates as well as annotations inserted by the NLM such as creation date of the Medline entry, list of chemicals associated with the document, as well as related MeSH headings.

The text processing modules are only concerned with the document parts that consist of natural language text. In Medline abstracts, these stretches of text are marked up as *ArticleTitle* and *AbstractText*. Inside these elements we add another XML element, called *text*, to flag natural language text independent of the original input document format (module M-1 in Figure 1). Thereby the subsequent text processing modules become independent of the document structure: other document types, e.g. BioMed Central

full text papers, can easily be fed into the pipeline providing a simple adaptation of the input pre-processor.

As a final pre-processing step (M-2), sentences are identified and marked using the <SENT> tag.

3.3 Finding protein names in text

For identification of protein names (M-3 in Figure 1), we use an existing protein repository. UniProt/Swiss-Prot contains roughly 190,000 protein/gene names (PGNs) in database entries that also annotate proteins with protein function, species and tissue type. PGNs from UniProt/Swiss-Prot are matched with regular expressions which account for morphological variability. These terms are tagged using the <z:uniprot> tag (see Figure 2). The list of identifiers (*ids* attribute) contains the accession numbers of the mentioned protein in the UniProt/Swiss-Prot database. All synonyms from a database entry are kept, and in the case of homonymy, where one name refers to several database entries, all accession numbers are stored. The pair consisting of the database name and the accession number(s) forms a unique identifier (UID) that represents the semantics of the term and can be trivially rewritten into a URL pointing to the database entry. Each entity also contains the attribute *fb* which provides the frequency of the term in the British National Corpus (BNC).

3.4 Resolving (some) protein name ambiguities

The approach to finding names that we presented can create three types of ambiguities mentioned above in Section 2.

In the current implementation, **Amb1** (ambiguity within a given resource) is not resolved. Rather, the links to *all* entries in the same database are maintained. **Amb2** and **Amb3** are partially resolved for protein/gene names as explained below (steps M-4 and M-5). Note that **Amb2** is resolved on "first-come first-serve" basis, meaning that an annotation introduced by one module is not overwritten by a subsequent module.

Many protein names are indeed or at least look like abbreviations. It has been proved that ambiguities of abbreviations and acronyms found in Medline abstracts can be automatically resolved with high accuracy (Yu et al., 2002; Schwartz and Hearst, 2003; Gaudan et al., 2005).

```
<SENT sid="2" pm="."> Aberrant
Wnt signaling, which results from
mutations of either <z:uniprot
fb="0" ids="P26233,P35222,P35223,
P35224,Q02248, Q9WU82">beta-
catenin</z:uniprot> or adenomat-
ous polyposis coli (<z:uniprot
fb="28" ids="P25054">APC </z:uni-
prot>), renders <z:uniprot fb="0"
ids= "P26233,P35222, P35223,
P35224,Q02248, Q9WU82"> beta-
catenin</z:uniprot> resistant to
degradation, and has been associ-
ated with multiple types of human
cancers
</SENT>
```

Figure 2. XML annotation of UniProt/Swiss-Prot proteins .

In our approach (Gaudan et al., 2005) all acronyms from Medline have been gathered together
with their expanded forms, called senses. In addition all morphological and syntactical variants of a known expanded form have been extracted from Medline. Expanded forms were categorised into classes of semantically equivalent forms. Feature representations of Medline abstracts containing the acronym and the expanded form were used to train support vector machines (SVMs). Disambiguation of acronyms to their senses in Medline abstracts based on the SVMs was achieved at an accuracy of above 98%. This was independent from the presence of the expanded form in the Medline abstract. This disambiguation solution lead to the solution integrated into the processing pipeline.

A potential protein has to be evaluated against three possible outcomes: either a name is an acronym and can be resolved as (a) a protein or (b) not a protein, or (c) a name cannot be resolved. To distinguish cases (a) and (b) the document content is processed to identify the expanded form of the acronym and to check whether the expanded form refers to a protein name. In case of (c), the frequency of the name in the British National Corpus (BNC) is compared with a threshold. If the frequency is higher than the threshold, the name is assumed not to be a protein name. The threshold was chosen not to exclude important protein names that have already entered common English (such as *insulin*).

The disambiguation module (M-4) runs on the results of the previous module that performs protein-name matching and indiscriminately assumes each match to be a protein name. The

module M-4 marks up all known acronym expansions in the text and combines the two pieces of information: a marked up protein name is looked up in the list of abbreviations. If the abbreviation has an expansion that is marked up in the vicinity **and** denotes a protein name, the abbreviation is verified as a protein name (case (a) above) by adding an attribute with a suitable value to the protein tag. The annotation also includes the normalised form of the acronym, which serves as an identifier for further database lookups. Similarly, if the expansion is clearly not a protein name, the same attribute is used with the according value.

Finally, the module M-5 removes the protein name markup if the name is either (b) clearly not a protein, or in case (c) has a BNC frequency beyond the threshold.

3.5 Finding other names in text

Further modules (M-6, M-7 and M-8 in Fig. 1) perform matching and markup for drugs from MedlinePlus[9], species from Entrez Taxonomy[10] and terms from the Gene Ontology (GO). As for proteins, the semantic type is signified by the element name and a unique ID referencing the source database is added as an attribute. Disambiguation for these names and terms is, however, not yet available.

Finding GO ontology terms in text can be difficult, as these names are typically "descriptions" rather than real terms (e.g. GO:0016886, *ligase activity, forming phosporic ester bonds*), and therefore are not likely to appear in text frequently (McCray et al., 2002; Verspoor et al., 2003; Nenadic et al., 2004).

Figure 3 shows an example of a sentence annotated for semantic types and POS information using the pipeline from the Figure 1. Note that POS tags are inside the type tags although type annotation has been performed prior to the POS tagging.

3.6 Other modules in the pipeline

The modular text processing pipeline of EBIMed is currently being extended to include other modules. The part-of-speech tagger (POS-tagger) is a separate module and combines tokenization and POS annotation. It leaves previously annotated entities as single tokens, even for multi-word terms, and assigns a noun POS tag to every named entity.

Shallow parsing is introduced as another layer in the multidimensional annotation of biomedical documents. After the NER modules, the shallow parsing modules extract events of protein-protein interactions. Shallow parsing basically annotates noun phrases (NP) and verb groups. Noun phrases that contain a protein name receive a modified NP tag (Protein-NP) to simplify finding of protein-protein interaction phrases. Patterns of Protein-NPs in conjunction with selected verb groups are annotated as final result.

```
<abs id='1' db='unknown'>
<text><SENT sid="0" pm="."><tagged>
<tok><sur> </sur><lem cat="bos"
mor=""></lem></tok>
<z:uniprot fb="0" ids="P50144,P50145">
<tok><sur>Cholecystokinin</sur><lem
cat="n" mor=":e">cholecystokinin</lem>
</tok> </z:uniprot>
<tok><sur>and</sur><lem cat="cnj"
mor=":K">and</lem></tok>
<z:uniprot fb="4" ids="O02686,P01350">
<tok><sur>gastrin</sur><lem cat="n"
mor=":e">gastrin</lem></tok>
</z:uniprot>
<tok><sur>differed</sur><lem cat="v"
mor=":V:P">differ</lem></tok>
<tok><sur>in</sur><lem cat="prep"
mor="">in</lem></tok>
<tok><sur>stimulatin</sur><lem cat="n"
mor=":e:m">stimulatin</lem></tok>
<z:uniprot fb="4" ids="O02686,P01350">
<tok><sur>gastrin</sur><lem cat="n"
mor=":e">gastrin</lem></tok>
</z:uniprot>
<z:go ids="GO:0046903"
onto="biological_process">
<tok><sur>secretion</sur><lem cat="n"
mor=":e">secretion</lem></tok>
</z:go>
<tok><sur>in</sur><lem cat="prep"
mor="">in</lem></tok>
<z:species ids="9986">
<tok><sur>rabbit</sur><lem cat="n"
mor=":e">rabbit</lem></tok>
</z:species>
<tok><sur>gastric</sur><lem cat="adj"
mor=":b">gastric</lem></tok>
<tok><sur>glands</sur><lem cat="n"
mor=":m">gland</lem></tok>
<tok><sur>.</sur><lem cat="eos"
mor=""></lem></tok>
</tagged></SENT>
</text>
</abs>
```

Figure 3. XML annotation of a sentence containing different semantic types and POS tags.

[9] MedlinePlus, National Library of Medicine, http://www.nlm.nih.gov/medlineplus/

[10] Entrez Taxonomy, National Center for Biotechnology Information, http://www.ncbi.nlm.nih.gov/entrez/

4 EBIMed

This cascaded approach to NER has been incorporated into EBIMed, a system for mining biomedical literature.

EBIMed is a service that combines document retrieval with co-occurrence-based summarization of Medline abstracts. Upon a keyword query, EBIMed retrieves abstracts from EMBL-EBI's installation of Medline and filters for biomedical terminology. The final result is organised in a view displaying pairs of concepts. Each pair co-occurs in at least one sentence in the retrieved abstracts. The findings (e.g. UniProt/Swiss-Prot proteins, GO annotations, drugs and species) are listed in conjunction with the UniProt/Swiss-Prot protein that appears in the same biological context. All terms, retrieved abstracts and extracted sentences are automatically linked to contextual information, e.g. entries in biomedical databases.

The annotation modules are also available via HTTP request that allows for specification of which modules to run (cf. Whatizit[11]). Note that with suitable pre-processing to insert the `<text>` tags, even well formed HTML can be processed.

5 Lessons Learnt so far

Our text mining solution EBIMed successfully applies multi-dimensional markup in a pipeline of text processing modules to facilitate online retrieval and mining of the biomedical literature. The final goal is semantic annotation of biomedical terms with UID, and – in the next step – shallow parsing based text processing for relationship identification. The following lessons have been learnt during design, implementation and use of our system.

The end-users expect to see the original document at all times and therefore we have to rely on proper formatting of the original and the processed text. Consequently, when adding semantic information, all other meta-information must be preserved to allow for proper rendering as similar as possible to the original document. Therefore, our approach does not remove any pre-existing annotations supplied by the publisher, i.e. the original document could be recovered by removing all introduced markup.

All modules only process sections of the document containing the natural language text, which improves modularisation. The document structure is irrelevant to single modules and facilitates reading and writing to the input and output stream, respectively, without taking notice of the beginning and/or the end of a single document. All information exchanged between modules is contained in the data stream. This facilitates running all the modules in a given pipeline in parallel, after an initial start-up. Even more, the modules can be distributed on separate machines with no implementation overheads for the communication over the network. Adding more modules with their own processors does not significantly impair overall runtime behaviour for large datasets and leads to fast text processing throughput combined with a reasonable — albeit not yet perfect — quality, which allows for new and practically useful text mining solutions such as EBIMed.

Modularisation of the text processing tasks leads to improved scalability and maintainability inherent to all modular software solutions. In the case of the presented solution, the modular approach allows for a selection of the setup and ordering of the modules, leading to a flexible software design, which can be adapted to different types of documents and which allows for an (incremental) replacement of methods to improve the quality of the output. This can also facilitate improved interoperability of XML-based NLP tools.

Semantic annotation of named entities and terms blends effectively with logical markup, simply because there is no overlap between document structure and named entities and terms. On the other hand, some physical markup (such as <*i*> in the BMC corpus) is in some documents used to highlight names or terms of a semantic type, e.g. gene names. With consistent semantic markup, this kind of physical tags could be abandoned to be replaced by external style information. However, some semantic annotations still must be combined with physical markup as in the term ***B-****sup* that initially was annotated by a publisher as *B-sup* and that now (after NER) would be marked as `<z:uniprot><b>B</b>-sup</z:uniprot>`.

Matching of names of a semantic type, e.g. protein/gene, is done on a "longest of the leftmost" basis and prioritization of semantic types is enforced by the order of the term identification modules. Both choices lead to the result that overlapping annotations are preempted and that annotations automatically endorse a link to a unique identifier, unless there are ambiguity on the level of biomedical resource.. This type of ambiguity is not resolved in our text processing solution. Instead, for a given biomedical term, links to all entries referring to this term in the same database are kept.

[11] http://www.ebi.ac.uk/Rebholz-srv/whatizit/pipe

One approach to the disambiguation of **Amb2** (multiple resources) and **Amb3** (common English words) ambiguities would be to integrate all terms into one massive dictionary, identify the strings in the text and then disambiguate between *n* semantic types. This would require the disambiguation module be trained to distinguish all semantic types. If a new type is added, the disambiguation module would need to be retrained, which limits the possibilities for expansion and tailoring of text mining solutions.

Open Problems: We consider two categories of open problems: *NLP-based* and *XML-based* problems.

Bio NLP-based problems include challenges in recognition and disambiguation of biomedical names in text. One of the main issues in our approach is annotation of compound and nested terms. The presented methodology can lead to the following annotations:

1. the head noun belongs to the same semantic type, but is not part of the protein name (as represented in the terminological resource):
   `<z:uniprot>Wnt-2</z:uniprot> protein`
2. the head noun belongs to a different semantic type not covered by any of the available terminological resources:
   `<z:uniprot>WNT8B</z:uniprot> mRNA`
3. a compound term consists of terms from different semantic types, but its semantic type is not known:
   `<z:uniprot fb="0" ids="…">beta-catenin</z:uniprot> <z:go ids="…" onto= "…">binding </z:go> domain`

Therefore, an important open problem is the annotation of nested terms where an entity name is part of a larger term that may or may not be in one of the dictionaries. Once the inner term is marked up with inline annotation, simple string pattern matching (utilised in our approach) cannot be used easily to find the outer, because the XML structure is in the way. A more effective solution could be a combination of inline with stand-off annotation.

Further, in a more complex case such as in

`htr-wnt-<uniprot>A protein</uniprot>`

neither `wnt` nor `htr` refer to a single protein but to a protein family, and whereas `A protein` is a known protein, this is not the case for `wnt-A`. The most obvious annotation `<uniprot>htr-wnt-A protein</uniprot>` cannot be resolved by the terminology from the UniProt/Swiss-Prot database, as it simply does not exist in the database.

More work is also needed on disambiguation of terms that correspond to common English words.

Annotation (i.e. XML)-based problems mainly relate to an open question whether different tag names should be used for various semantic types, or semantic types should be represented via attributes of a generalised *named entity or term* tag. In EBIMed, specific tags are used to denote specific semantic types. A similar challenge is how to treat and make use of entities such as inline references, citations and formulas (typically annotated in journals), which are commonly ignored by NLP modules.

The most important issue, however, is how to represent still unresolved ambiguities, so that annotations might be modified at a later stage, e.g. when POS information or even the full parse tree is available. This also includes the issues on kind of information that should be made available for later processing. For example, as (compound) term identification is done before POS tagging, an open question is whether POS information should be assigned to individual components of a compound term (in addition to the term itself), since this information could be used to complete NER or adjust the results in a later stage.

6 Conclusions

In this paper, we have described a pipeline of XML-based modules for identification and disambiguation of several semantic types of biomedical named entities. The pipeline processes and semantically enriches documents by adding, changing or removing annotations. More precisely, the documents are augmented with UIDs referring to referential databases. In the course of the processing, the number of annotated NEs increases and the quality of the annotation improves. Thus, one of the main issues is to represent still unresolved ambiguities consistently, so that the following modules can perform both identification and disambiguation of new semantic types. As subsequent modules try to add new semantic annotations, prioritization of semantic types is enforced by the order of the term identification modules.

We have shown that such approach can be employed in a real-world, online information mining system EBIMed. The end-users expect to view the original layout of the documents at all times, and thus the solution needs to provide an efficient multidimensional markup that preserves and combines existing markup (from publishers) with semantic NLP-derived tags. Since, in the biomedical domain, it is essential to provide

links from term and named-entity occurrences to referential databases, EBIMed provides identification and disambiguation of such entities and integrates text with other knowledge sources.

The existing solution to annotate only longest non-overlapped entities is useful for real world use scenarios, but we also need ways to improve annotations by representing nested and overlapped terms.

Acknowledgements

The development of EBIMed is supported by the Network of Excellence "Semantic Interoperability and Data Mining in Biomedicine" (NoE 507505). Medline abstracts are provided from the National Library of Medicine (NLM, Bethesda, MD, USA) and PubMed is the premier Web portal to access the data.

Sylvain Gaudan is supported by an "E-STAR" fellowship funded by the EC's FP6 Marie Curie Host fellowship for Early Stage Research Training under contract number MESTCT- 2004-504640. Goran Nenadic acknowledges supported from the UK BBSRC grant "Mining Term Associations from Literature to Support Knowledge Discovery in Biology" (BB/C007360/1).

EBI thanks IBM for the grant of an IBM eServer BladeCenter for use in its research work.

References

A. Bairoch, R. Apweiler, C.H. Wu, W.C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, M.J. Martin, D.A. Natale, C. O'Donovan, N. Redaschi and L.S. Yeh. 2005. The Universal Protein Resource (UniProt). *Nucleic Acids Research,* 33(Database issue):D154-9.

C. Blaschke, M.A. Andrade, C. Ouzounis and A. Valencia. 1999. Automatic extraction of biological information from scientific text: Protein-protein interactions. *Proc. ISMB*, 7:60–7.

A.C. Browne, G. Divita, A.R Aronson and A.T. McCray. 2003. UMLS language and vocabulary tools. *AMIA Annual Symposium Proc.*, p. 798.

L. Chen, H. Liu and C. Friedman. 2005. Gene name ambiguity of eukaryotic nomenclature. *Bioinformatics*, 21(2):248-56

S. Gaudan, H. Kirsch and D. Rebholz-Schuhmann. 2005. Resolving abbreviations to their senses in Medline. *Bioinformatics,* 21(18):3658-64

GO Consortium. 2006. The Gene Ontology (GO) project in 2006. *Nucleic Acids Research*, 34(suppl_1):D322-D326.

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.

H. Kirsch, S. Gaudan and D. Rebholz-Schuhmann. 2005. Distributed modules for text annotation and IE applied to the biomedical domain. *International Journal Medical Informatics*. (doi:10.1016/j.ijmedinf.2005.06.011)

M. Krauthammer and G. Nenadic. 2004. Term identification in the biomedical literature. *Journal Biomedical Informatics*, 37(6):512-26.

K. Lee, Y. Hwang, and H. Rim. 2003. Two-Phase Biomedical NE Recognition based on SVMs. *Proc. of NLP in Biomedicine*, *ACL 2003*. p. 33-40.

H. Liu, S.B. Johnson, and C. Friedman, 2002. Automatic resolution of ambiguous terms based on machine learning and conceptual relations in the UMLS. *J Am Med Inform Assoc*, 2002. 9(6): p. 621-36.

A. McCray, A. Browne and O. Bodenreider O. 2002. The lexical properties of Gene ontology (GO). *Proceedings of AMIA 2002*. 2002:504-8.

G. Nenadic, I. Spasic, and S. Ananiadou. 2005. Mining Biomedical Abstracts: What's in a Term?, *LNAI* Vol. 3248, pp. 797-806, Springer-Verlag

G. Nenadic and S. Ananiadou. 2006. Mining Semantically Related Terms from Biomedical Literature. *ACM Transactions on ALIP*, 01/2006 (Special Issue Text Mining and Management in Biomedicine)

xD. Rebholz-Schuhmann, H. Kirsch, M. Arregui, S. Gaudan, M. Rynbeek and P. Stoehr. (forthcoming) Identification of proteins and their biological context from Medline: EBI's text mining service EBIMed.

D. Rebholz-Schuhmann, S. Marcel, S. Albert, R. Tolle, G. Casari and H. Kirsch. 2004. Automatic extraction of mutations from Medline and cross-validation with OMIM. *Nucleid Acids Research*, 32(1):135–142.

A. Rzhetsky, I. Iossifov, T. Koike, M. Krauthammer, P. Kra, et al. 2004. GeneWays: A system for extracting, analyzing, visualizing, and integrating molecular pathway data. *Journal Biomedical Informatics*, 37:43–53.

A.S. Schwartz and M.A. Hearst. 2003. A simple algorithm for identifying abbreviation definitions in biomedical text. *Proceedings of Pac Symp Biocomput*. 2003. p. 451-62.

M. Torii, S. Kamboj and K. Vijay-Shanker. 2003. An Investigation of Various Information Sources for Classifying Biological Names. *Proceedings of NLP in Biomedicine, ACL 2003*. p. 113-120

CM Verspoor, C. Joslyn and G. Papcun. 2003. The Gene ontology as a source of lexical semantic knowledge for a biological natural language processing application. *Proc. of Workshop on Text Analysis and Search for Bioinformatics,* SIGIR 03

H. Yu, G. Hripcsak and C. Friedman. 2002. Mapping abbreviations to full forms in biomedical articles. *J Am Med Inform Assoc*, 2002. 9(3): p. 262-72.