Feeding OWL: Extracting and Representing the Content of Pathology Reports

David Schlangen and Manfred Stede
Department of Linguistics
University of Potsdam
P.O. Box 601553
D-14415 Potsdam, Germany
{das|stede}@ling.uni-potsdam.de

Elena Paslaru Bontas
Institute for Computer Science
Freie Universität Berlin
Takustr.9
D-14195 Berlin, Germany
paslaru@inf.fu-berlin.de

Abstract

This paper reports on an ongoing project that combines NLP with semantic web technologies to support a content-based storage and retrieval of medical pathology reports. We describe the NLP component of the project (a robust parser) and the background knowledge component (a domain ontology represented in OWL), and how they work together during extraction of domain specific information from natural language reports. The system provides a good example of how NLP techniques can be used to populate the Semantic Web.

1 Introduction

Clinical pathologists work with and produce vast amounts of data: images of biological samples and written reports of their findings. *Digital Pathology* is the cover term for a number of efforts to introduce digital processing into the work-flow of the pathologist. While previous projects have focussed on storage and distribution of images and reports (e.g. in *Tele-Pathology*-projects, (Slodowksa et al., 2002; Demichellis et al., 2002)), the work reported here explores the use of Natural Language Processing (NLP) and Semantic Web technologies to support a *content-based* storage and retrieval of case reports. The system that we are building, LUPUS (Lung Pathology System), consists of an NLP component (a robust parser) and a Semantic Web component (a domain ontology represented in OWL, and a Description Logic reasoner), which work closely together, with the domain ontology guiding the information extraction process.

The remainder of the paper is organised as follows. In the next section we describe the context and intended application of the system, we discuss linguistic properties of the input material we are working with, and we give some details of the background ontology we are using. In Section 3 we go into the technical details of the process of extracting information from natural language reports and representing it in an OWL representation, after which we describe a preliminary evaluation. We close with discussing related work, and planned future work.

2 Digital Pathology

2.1 The Application

LUPUS is intended to support the pathologist in two ways. First, it is used to semantically annotate a large archive of case reports, turning them into a valuable resource for diagnosis and teaching. The system uses the case reports produced by experts (the pathologists) to extract information about the accompanying images (of the tissue samples), and thus produces semantic annotation both for the report and for those images.

This corpus of cases can then be searched in a fast, *content-based* manner to retrieve case reports (the textual reports together with the images of tissue samples) that might be relevant for a case the pathologist is working on. The search is content-based in that it can make use of semantic relationships between search concepts and those occuring in the text. We also encode in rules knowledge about certain diagnostics tasks, so that for example queries asking for 'differential diagnosis' ("show me cases of diagnoses which are known to be easily confusable with the diagnosis I am thinking of for the present case") can be processed—tasks which normally require consultation of textbooks. These search capabilities are useful both during diagnosis and for teaching, where it makes interesting examples immediately available to students.

Another use case is quality control during input of new reports. Using our system, such reports can be entered in a purpose-built editor (which combines digital microscopy facilities (Saeger et al., 2003) with our semantic annotator / search engine), where they are analysed on-the-fly, and potential inconsistencies with respect to the background domain ontology are spotted.[1] During the development phase of the system, we are using this feature

[1] Naturally, to gain acceptance by working pathologists, this process has to be "minimally invasive".

to detect where the coverage of the system must be extended.

The present paper focuses on the process of extracting the relevant information from natural language reports and representing it in a semantic web-ready format as a precondition for performing searches; we leave the description of the search and retrieval functions to another paper. To give an idea of the kind of data we are dealing with, and of the intended target representation, Figure 1 shows an example report (at the top of the figure) and the representation of its content computed by our system (at the bottom).[2] We discuss the input format in the following subsection, and the target representation together with the domain knowledge available to us in Subsection 2.3; discussion of the intermediate format that is also shown in the figure is deferred until Section 3.

2.2 Pathology Reports

During the development phase of the system, we are using a corpus of 90 randomly selected case reports (ca. 13,000 words; i.e. the average length of the reports is ca. 140 words, with a standard deviation of 12 words) for testing and grammar development. Linguistically, these reports are quite distinguished: they are written in a "telegram"-style, with verbs largely being absent (a rough examination of the corpus showed that only about every 43rd token is a verb, compared to every 11th in a comparable corpus of German newspaper). Also, the vocabulary is rather controlled, with very little variation—this of course is good news for automatically processing such input. On the discourse level we also find a strict structure, with a fixed number of semantically grouped sections. E.g., information about the diagnosis made will normally be found in the section "Kritischer Bericht" (critical report), and the information in the "Makroskopie" and "Mikroskopie" sections (macroscopy and microscopy, respectively) will be about the same parts of the sample, but on different levels of granularity.

The last peculiarity we note is the relatively high frequency of compound nouns. These are especially important for our task, since technical concepts in German tend to be expressed by such compound nouns (rather than by noun groups). While some of those will denote individual concepts and hence will be recorded in the domain lexicon, others must be analysed and their semantics must be composed out of that of their parts (see below).

2.3 Lung Pathology Knowledge in OWL

The result of processing such reports with LUPUS is a representation of (relevant aspects of) their content. This representation has the form of instances of concepts and assertions of properties that are defined in an ontology, which constitutes the domain knowledge of the system (at the moment focussed on pathologies of the lung). This ontology is specified in OWL DL (W3C WebOnt WG, 2004), a version of OWL with a formal semantics and a complete and decidable calculus. Consequently, the content of the texts is represented in OWD DL as well, and so the knowledge base of the system consists of the ontology and the instances.

The ontology we use is compiled out of several medical sources (such as UMLS (The UMLS Consortium, 2003) and SNOMED (SNOMED International, 2004)), but since these sources often were not intended for machine *reasoning* (i.e., are not necessarily consistent, and use rather loosely defined relations), considerable effort has been spent (and is being spent) on cleaning them up.[3] At the moment, about 1,000 domain-level concepts and ca. 160 upper-level concepts have been identified, which are connected by about 50 core relation types. To our knowledge, this makes it one of the biggest OWL-ontologies currently in use.

Besides representing concepts relevant to our domain, the ontology also lists properties that instances of these concepts can have. These properties are represented as two-place relations; to give an example, the property "green" attributed to an entity x will in our system not be represented as "green(x)", but rather as something like "colour(x, green)". This allows us to enforce consistency checks, by demanding that for each second-order predicate (colour, malignity, consistency, etc.) appropriate for a given concept only one value is chosen.[4] This choice of representation has consequences for the way the semantics of adjectives is represented in the lexicon, as we will see presently.

[2] What is shown in the figure is actually already the result of a preprocessing step; the cases as stored in the database contain patient data as well, and are formatted to comply with the HL7 standard for medical data (The HL7 Consortium, 2003). Moreover, the italicisation in the input representation and the numbers in square brackets are added here for ease of reference and are *not* part of the actual representations maintained by the system.

[3] There are several current research projects with a similar aim of extracting stricter ontologies from sources like those mentioned above (see e.g. (Schulz and Hahn, 2001; Burgun and Bodenreider, 2001)), and this is by no means a trivial task. The present paper, however, focuses on a different (but of course interdependent) problem, namely that of extracting information such that it can be represented in the way described here.

[4] Technically, these constraints are realised by functional data-properties relating entities to enumerated data types.

An example report (with translation):

<befund>
<makroskopie>
Stanzzylinder von 15 mm Länge und 1 mm Durchmesser. [1]
</makroskopie>
<mikroskopie>
Stanzbiopsat [2] eingenommen durch *Infiltrate einer soliden malignen epithelialen Neoplasie.* [3] *Die Tumorzellen mit distinkten Zellgrenzen* [4], zum Teil interzellulär *Spalträume* [5], zwischen denen stellenweise *kleine Brücken* [6] nachweisbar sind. *Das Zytoplasma leicht basophil, z.T. auch breit und eosinphil,* [7] *die Zellkerne hochgradig polymorph mit zum Teil multiplen basophilen Nukleolen.* [8] *Deutliche desmoplastische Stromareaktion.* [9]
</mikroskopie>
<kritischer_bericht>
Stanzbiopsat aus einer Manifestation eines soliden Karzinoms [10]
(klinisch rechte Lunge apikal).
</kritischer_bericht>
<kommentar>
...
</kommentar>
</befund>

(Biopsy cylinder of 15 mm length and 1 mm diameter. | Biobsy infiltrated by a solid malignant epithelial neoplasia. The tumor cells with distinct cell borders, partially intercellular spatia, between which sporadically small bridges are verifiable. The cytoplasm lightly basophil, in part also broad and eosinphile, the nuclei highly polymorphic, partially with multiple basophile nucleoli. Distinct desmoplastic stroma reaction. | Biopsy cylinder from a manifestation of a solid carcinoma (clinical right lung apical).)

⇓

Intermediate Representation (excerpt):

[2] $unspec\_det(x_2) \wedge punch\_biopsat(x_2)$ [3] $unspec\_plur\_det(x_3) \wedge infiltrate(x_3, x_4) \wedge indef\_det(x_4) \wedge solid(x_4) \wedge malign(x_4) \wedge epithelial(x_4) \wedge neoplasia(x_4)$
[4] $def\_plur\_det(x_5) \wedge tumorcell(x_5) \wedge with\_rel(x_5, x_6) \wedge unspec\_plur\_det(x_6) \wedge distinctive(x_6) \wedge cell\_borders(x_6)$ [7] $spec\_det(x_9) \wedge low\_degree(d_1) \wedge basophile(x_9, d_1) \wedge partially(d_2) \wedge broad(x_9, d_2) \wedge eosinphile(x_9, d_2) \wedge cytoplasm(x_9)$
[8] $def\_plur\_det(x_{10}) \wedge high\_degree(d_3) \wedge polymorpheous(x_{10}, d_3) \wedge nucleus(x_{10}) \wedge with\_rel(x_{10}, x_{11}) \wedge unspec\_plur\_det(x_{11}) \wedge partially(d_4) \wedge multiple(x_{11}, d_4) \wedge basophile(x_{11}) \wedge nucleoli(x_{11})$

⇓

Target Representation (excerpt):

<Malignant_Epithelial_Neoplasm_C0432650 rdf:ID="neoplasia_x4">
<solidity rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</solidity>
</Malignant_Epithelial_Neoplasm>
<Cell_Border_C0032743 rdf:ID="cell_border_x61"/>
<Tumor_cells_C0431085 rdf:ID="tumor_cell_x52">
<hasBoundary rdf:resource="file:...#cell_boundary_x61"/>
</Tumor_cells_C0431085>
<cytoplasm_C0326583 rdf:ID="cytoplasm1">
<broad rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</broad>
<eosinphil rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</eosinphil>
<basophil rdf:datatype="http://www.w3.org/2001/XMLSchema#float">0.5</basophil>
</cytoplasm>

Figure 1: Input, Intermediate and Target Representation

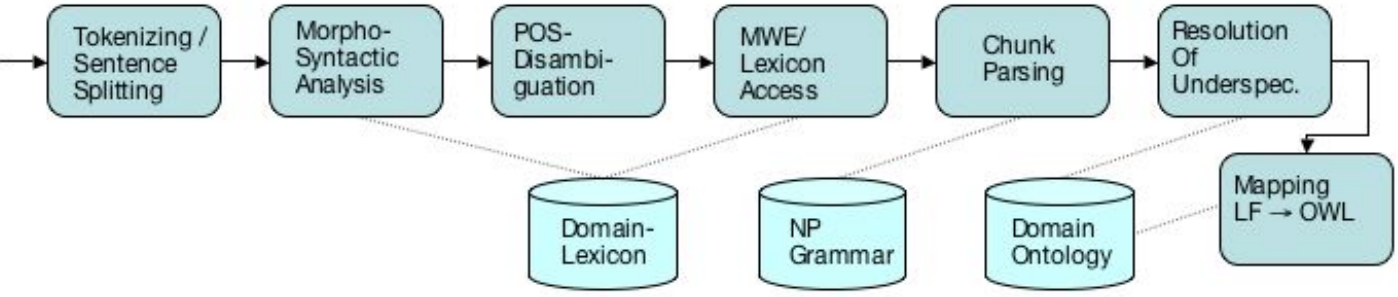

Figure 2: Flowchart

Using OWL DL as a representation format for natural language content means certain limitations have to be accepted. Being a fragment of FOL, it is not expressive enough to represent certain finer semantic details, as will be discussed below. However, the advantage of using an emerging standard for delivering and sharing information outweighs these drawbacks.

3 Implementation

3.1 Overview

As mentioned above, most of the sentences in our corpus do not contain a finite verb; i.e., according to standard rules of grammar they are *elliptical*. While a theoretically motivated approach should strive to resolve this ellipsis contextually (for example as described in (Schlangen, 2003)), in view of the intended application and for reasons of robustness we have decided to focus only on extracting information about the entities introduced in the reports—that is, on recognising *nominal phrases*, leaving aside the question of how verbal meanings are to be resolved.

Our strategy is to combine a "shallow" preprocessing stage (based on finite-state methods and statistical approaches) with a symbolic phase, in which the semantics of the NPs is assembled.[5] A requirement for the processing is that it must be *robust*, in two ways: it must be able to deal with unknown *tokens* (i.e., "out of vocabulary" items) and with unknown *structure* (i.e., "out of grammar" constructions), degrading gracefully and not just failing.

Figure 2 shows a flow chart of the system; the individual modules are described in the following sections.

[5]This strategy sits somewhere between *Information Extraction*, where also only certain phrases are extracted, for which, however, normally no compositional semantics is computed, and "full" parsing, where such a semantics is computed only if the whole input can be parsed.

3.2 Preprocessing

The first step, tokenising and sentence splitting, is fairly standard, and so we skip over it here. The second step, morpho-syntactic analysis, is more interesting. It is performed by an independently developed module called TAGH, a huge finite-state machine that makes use of a German word-stem lexicon (containing about 90,000 entries for nouns, 17,000 for verbs, 20,000 adjectives and adverbs, and about 1,500 closed class word forms). The transducer is implemented in C++ and has a very high throughput (about 20,000 words per second on modern machines). The coverage achieved on a balanced corpus of German is around 96% (Jurish, 2003), for our domain the lexicon had to be extended with some domain specific vocabulary.

To give an example of the results of the analysis, Figure 3 shows (excerpts of) the output for Sentence 2 of the example report. Note that this is already the POS-disambiguated output, and we only show one analysis for each token. In most cases, we will get several analyses for each token at this stage, differing with respect to their part of speech tag or other morphological features (e.g., case) that are not fully determined by their form. (The average is 5.7 analyses per token.) Note also that the actual output of the module is in an XML format (as indeed are all intermediate representations); only for readability is it presented here as a table.

Another useful feature of TAGH is that it provides derivational information about compound nouns. To give an example, (1) shows one analysis of the noun "Untersuchungsergebnis" (examination result).

(1) Untersuchungsergebnis
untersuch(V)∼ung(n)/s#Ergebnis

As this shows, the analysis gives us information about the stems of the compounds; this can be used to guide the computation of the meaning of the complex noun. However, this meaning is not fully com-

Token	Type	Analysis
Stanzbiopsat	Stanzbiopsat	[NN Gender=neut Number=sg Case=nom]
eingenommen	ein|nehm∼en	[VVPP2]
durch	durch	[APPR]
Infiltrate	Infiltrat	[NN Gender=neut Number=pl Case=acc]
einer	eine	[ARTINDEF Number=sg Case=gen Gender=fem]
soliden	solid	[ADJA Degree=pos Number=sg Case=gen Gender=* ADecl=mixed]
malignen	maligne	[ADJA Degree=pos Number=sg Case=gen Gender=* ADecl=mixed]
epithelialen	epithelial	[ADJA Degree=pos Number=sg Case=gen Gender=* ADecl=mixed]
Neoplasie	Neoplasie	[NN Gender=fem Number=sg Case=*]

Figure 3: Result of Morphological Analysis / POS-tag disambiguation for Sentence 2

positional, as the nature of the relation between the compounds is *underspecified*. We represent this by use of an underspecified relation rel that holds between the compounds, and which has to be specified later on in the processing chain.

The output of this module is then fed into a statistically trained POS-disambiguator, which finds the most likely path through the lattice of morphological analyses (Jurish, 2003) (with an accuracy of 96%). In cases where morphology failed to provide an analysis, the syntagmatically most likely POS tag is chosen. At the end of this stage all analyses for a given token agree on its part of speech; however, other features (number, person, case, etc.) might still not be disambiguated.

At the next stage, certain sequences of tokens are grouped together, namely *multi-word expression* that denote a single concept in our ontology (e.g., "anthrakotische Lymphknoten" denotes a single concept, and hence is marked as one token of type NN at this step), and certain other phrases (e.g. specifications of spatial dimensions) which can be recognised easily but would require very specialised grammar rules later on.[6]

Then, the domain-specific lexicon is accessed, which maps "concept names" (nouns, or phrases as recognised in the previous step) to the concept IDs used in the ontology.[7] Tokens for which there is no entry in that lexicon, and which are hence deemed 'irrelevant' for the domain, are assigned a 'dummy' semantics appropriate for their part of speech, so that they do not confuse the later parsing stage. (More details about this kind of robustness will be given shortly.)

[6]See for example (Grover et al., 2002) for a discussion of the utility of a named entitiy recognition preprocessing stage for robust symbolic parsing.

[7]Note that this lexicon is one single resource out of which also the domain specfic additions to the morphology-lexicon and the list of multi-word expressions are compiled.

3.3 Chunk Parsing

Next, the analyses of the tokens are transformed into a feature structure format, and are passed to the parsing component.[8] The output of this stage is an intermediate semantic representation of (aspects of) the content (of which the notation shown in 1 is a variant). This format is akin to traditional logical forms and still has to be mapped into OWL; we decided on this strategy because such a format is closer to surface structure and hence easier to build compositionally (see discussion below in Section 3.5). Also note that the semantics is "flat", and does not represent scope of quantifiers (which only very rarely occur in our data, and cannot be represented OWL in any case).

To get an idea of the feature geometry used by the grammar see Figure 4; this figure also shows the semantic representations generated at this stage (in a different notation than in Figure fig:reps). Note the 'simulation' of typing of feature structures, and the representation of properties via second order properties as discussed above. Chunk parsing is performed by a chart parser running a grammar that is loosely inspired by HPSG (Pollard and Sag, 1994).[9] The grammar contains context-free rules for fairly complex NPs (allowing arguments of Ns, modification by PPs, and coordination). When extracting chunks, the strategy followed by the system is to always extract the largest non-overlapping chunks.[10]

An example might help to illustrate the robust-

[8]Up until here, all steps are performed in one go for the whole document. The subsequent steps, on the other hand, are performed incrementally for each sentence. This allows the system to remove ambiguity when it occurs, rather than having to maintain and later filter out different analyses.

[9]The parser is implemented in PROLOG, and based on the simple algorithm given in (Gazdar and Mellish, 1989). It also uses code by Michael Covington for dealing with feature structures in PROLOG, which is described in (Covington, 1994).

[10]That strategy will prefer lenght of individual chunks over coverage of input, for example when there is one big chunk and two overlapping smaller chunks at each side of that chunk, that however together span more input.

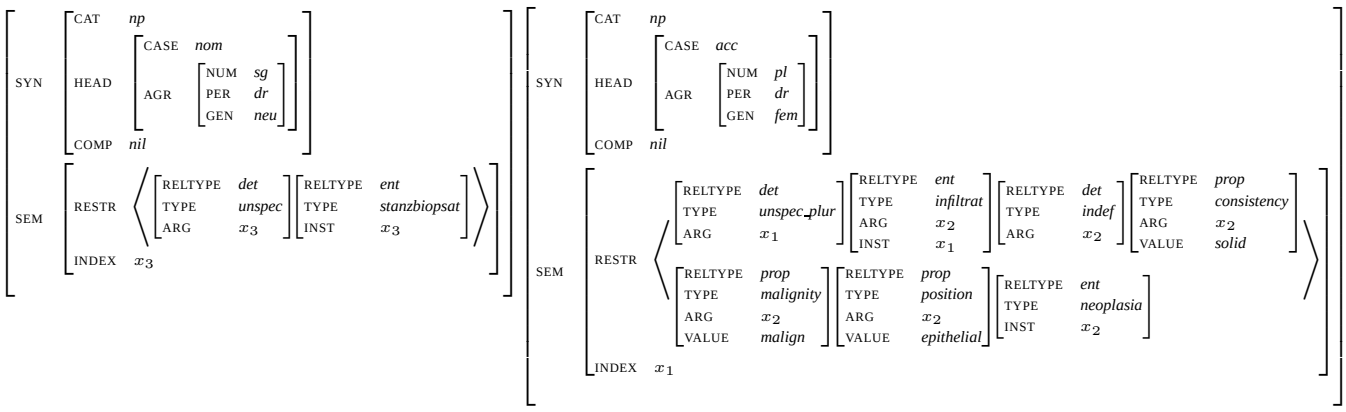

Figure 4: The chunks extracted from Sentence 2

ness of the system. (2) shows a full syntactic analysis of our example sentence. Our system only recognises the chunks indicated by the brackets printed in bold typeface: since it can't recognise the predicative use of the verb here, it is satisfied with just building parses for the NPs it does recognise. (The round brackets around the analysis of the first word indicate that this parse is strictly speaking not correct if the full structure is respected.)

(2) [NP **([NP)** [NOM Stanzbiopsat] **(])**, [ADJP [VVPP2 eingenommen] [PP [P durch] **[NP** Infiltrate einer soliden malignen epithelialen Neoplasie.**]**]]]"

This is an example of the system's tolerance to unknown *structure*; (3) shows a (constructed) example of an NP where the structure is covered by the grammar, but there are 'unknown' (or rather, irrelevant) *lexical items*. As described above, we assign a 'dummy semantics' (here, a property that is true of all entities) to words that are irrelevant to the domain, and so parsing can proceed.

(3) Solid, *hardly detectable* tumor cells. → $solid(x) \wedge true(x) \wedge tumor\_cell(x)$

A few last remarks about the grammar. First, as shown in Figure 4, NPs without determiner introduce an underspecified relation $unspec\_det$, and information about definiteness and number of determiners is represented. This means that all information to do discourse processing (bridging of definites to antecedents) is there; we plan to exploit such information in later incarnations of the system. Secondly, it can of course occur that there is more than one analysis spanning the same input; i.e., we can have syntactic ambiguity. This will be dealt with in the transformation component, where domain knowledge is used to only let through "plausible" analyses.

Lastly, prepositions are another source for underspecification. For instance, given as input the string (4), the parser will compute a semantics where an underspecified $with\_rel$ connects the two entities $tumor$ and $alveolar$; this relation will be specified in the next step, using domain knowledge, to a relation $contains$.

(4) Ein Tumor mit freien Alveolaren.
A tumor with free alveolars.

3.4 Resolution of Underspecification using Ontologies

As described in the previous sections, the output of the parser (and of the morphological analysis) might still contain underspecified relations. These are resolved in the module described in this section. This module sends a query to a reasoning component that can perform inference over the ontology, asking for possible relations that can hold between (instances of) entities. For example (4) above, this will return the answer $contains$, since the ontology specifies that 'alveolars" are parts of tumours (via a chain of *is-a*-relations linking tumours with cells, and cells with alveolars). In a similar way the underspecification of compound nouns is resolved. This process proceeds recursively, "inside-out", since compound nouns can of course be embedded in NPs that are parts of PPs, and so on.

3.5 Mapping LF to OWL

In the final step, the logical forms produced by the parser and specified by the previous module are transformed into OWL-compliant representations. This process is fairly straightforward, as should be clear from comparing the intermediate representation in Figure 1 with the target representation: a) unique identifiers for the instances of concepts are

generated; b) in cases of plural entities ("three samples" $\rightarrow card(x, 3) \wedge sample(x)$), several separate instances are created; and c) appropriateness conditions for properties are applied: if a property is not defined for a certain type of entity, the analysis is rejected.

This translation step also handles potential syntactic ambiguity, since it can filter out analyses if they specify inconsistent information. Note also that certain information, e.g. about second order properties, might be lost, due to the restricted expressivity of OWL. E.g., an expression like "highly polymorpheous" in Figure 1 either has to be converted into a representation like $polymorphism : high$, or the modification is lost ($polymorpheous(x)$).

This ends our brief description of the system. We now discuss a preliminary evaluation of the modules, related work, and further extensions of the system we are currently working on or which we are planning.

4 Evaluation

At the moment, we have only evaluated the modules individually, and—since the system is still under developement—this evaluation only provides a snapshot of the current state of developement. A full-scale evaluation of the whole system in its application context is planned as soon as the modules are finalised; plans for this are discussed below.

The coverage of the morphology module and the POS-tagger have already been reported above, so we concentrate here on the chunk-parser. To evaluate this module, we have manually annotated the NPs in a randomly selected test set of 20 reports (ca. 2,800 words; we found about 500 NPs). The reports were then morphologically analysed and POS-filtered, and the results were manually checked and corrected, to ensure that the input was optimal and really only the performance of the chunker was evaluated. We then computed precision and recall based on two different matching criteria: for *exact matching*, where only exact congruence of chunks counts, a precision of 48% and a recall of 63% was computed; the numbers improve when *partial matches*, i.e. smaller chunks within the target chunk, receive partial credit (by a factor of .25), resulting in a (relaxed) precision of 61% and a (relaxed) recall of 80%. This difference can be explained by the fact that some of the more complex NP-constructions (with quite complex modifications) in our data are not yet covered by the grammar, and only their constituent NPs are recognised.

Note that this evaluation just takes into account the boundaries of the chunks and not the correctness of the computed semantic representations. For a full-scale evaluation, we will manually annotate these NPs with semantic representations, and we will use this to compute precision and recall also with respect to semantics, and ultimately with respect to sample search queries. This annotation, however, is very resource-intensive, and so will only be done once the modules have been finalised.

5 Related Work

Acquisition of information from texts especially from the medical domain is a lively research area. Among the many projects in that field, we share some of our central concerns with the *medSynDiKAte* system (Hahn et al., 2002): robust text analysis of medical reports; a background knowledge base for guiding the analysis and storing the text's content; emphasis on handling co-reference phenomena. What distinguishes LUPUS from medSyndiKAte, though, is foremost the parsing scheme: the language used in the reports analysed by Hahn et al. is much closer to 'natural' language in that it contains sentences with tensed verbs. Accordingly, they use a variant of dependency parsing which is driven by verb information. As described in Section 2.2 above, this is not an option for us, given the style of our input texts, and hence our data renders a bottom-up chart parsing approach much more promising. Besides this difference, the work in medSynDiKAte predates the emergence of XML/web ontology standards and thus uses an earlier description logic knowledge representation language; we are hoping that by using a standard we will be able to allow even future semantic web technologies to work with our data.

As for the robust analysis side, (Grover et al., 2002), also use a similar preprocessing pipeline in combination with parsing. However, they also focus on more "natural" input texts (Medline abstracts), and they use statistical rather than symbolic/ontology based methods for computing the meaning of compound nouns.

6 Summary and Further Work

We have described LUPUS, an NLP system that makes use of a domain ontology to guide extraction of information about entities from medical texts, and represents this information as instances of concepts from that ontology. Besides its direct use for content-based search on these texts, the fact that the system relies entirely on emerging semantic web standards will make the resulting annotated information usable for all kinds of agents working with

such data.

As a next step, we plan to add *discourse processing* to the pipeline (see e.g. (Hahn et al., 1998) for a discussion why such a step is required even for such relatively simple texts). As mentioned above, the prerequisite information (about definite articles, for example) is already there; we plan to use the available domain knowledge to guide the search for antecedents for bridging. As a more technical improvement we are investigating ways of making the architecture less pipeline-y, and to integrate domain reasoning in computing edges in the chart. Lastly, we are also working on a large-scale evaluation of the system, by manually annotating reports to compute precision and recall.

Acknowledgements

We thank the anonymous reviewers for their helpful comments. Thanks are also due to Thomas Hanneforth and Bryan Jurish for their help with integrating their modules, and to our student assistant Sebastian Maar for doing much of the actual coding.

References

Anita Burgun and Oliver Bodenreider. 2001. Mapping the UMLS semantic network into general ontologies. In *Proceedings of the AMIA Symposium*.

Michael A. Covington. 1994. GULP 3.1: An extension of prolog for unification-based grammar. Technical Report AI-1994-06, University of Georgia.

F. Demichellis, V. Della Mea, S. Forti, P. Dalla Palma, and C.A. Beltrami. 2002. Digital storage of glass slide for quality assurance in histopathology and cytopathology. *Telemedicine and Telecare*, 8(3):138–142.

Gerald Gazdar and Chris Mellish. 1989. *Natural Language Processing in PROLOG*. Addison-Wesley, Wokingham, England.

Claire Grover, Ewan Klein, Mirella Lapata, and Alex Lascarides. 2002. XML-based NLP tools for analysing and annotating medical language. In *Proceedings of the 2nd Workshop on NLP and XML*, Taipei, Taiwan, September.

Udo Hahn, Martin Romacker, and Stefan Schulz. 1998. Why discourse structures in medical reports matter for the validity of automatically generated text knowledge bases. In *MedInfo '98 – Proceedings of the 9th World Congress on Medical Informatics*, pages 633–638, Seoul, Korea, August.

Udo Hahn, Martin Romacker, and Stefan Schulz. 2002. Creating knowledge repositories from biomedical reports: The medsyndikate text mining system. In *Pacific Symposium on Biocomputing*, pages 338–349, Hawai, USA, January.

Bryan Jurish. 2003. Part-of-speech tagging with finite state morphology. In *Proceedings of the Workshop on Collocations and Idioms: Linguistic, Computational and Psycholinguistic Perspectives*, Berlin, Germany, September.

Carl Pollard and Ivan Sag. 1994. *Head-Driven Phrase Structure Grammar*. CSLI / The University of Chicago Press, Chicago and London.

Kai Saeger, Karsten Schlüns, Thomas Schrader, and Peter Hufnagl. 2003. The virtual microscope for routine pathology based on a pacs system for 6 gb images. In *Proceedings of the 17th International Congress on Computer Assisted Radiology and Surgery (CARS)*, pages 299–304, London, UK, June.

David Schlangen. 2003. *A Coherence-Based Approach to the Interpretation of Non-Sentential Utterances in Dialogue*. Ph.D. thesis, School of Informatics, University of Edinburgh, Edinburgh, UK.

Stefan Schulz and Udo Hahn. 2001. Medical knowledge engineering–converting major portions of the umls into a terminological knowledge base. *International Journal of Medical Informatics*.

J. Slodowksa, K. Kayser, and P. Hasleton. 2002. Teleconsultation in the chest disorders. *European Journal for Medical Research*, 7(Suppl.I):80.

SNOMED International. 2004. SNOMED clinical terms. http://www.snomed.org/index.html.

The HL7 Consortium. 2003. HL7 version 2.5 ANSI standard, June. http://www.hl7.org.

The UMLS Consortium. 2003. UMLS release 2003AC. http://www.nlm.nih.gov/research/umls/.

W3C WebOnt WG. 2004. OWL web ontology language overview. W3C recommendation, W3C, February. http://www.w3.org/TR/2004/REC-owl-features-20040210/.