Ontology Engineering for the Semantic Annotation of Medical Data

Elena Paslaru Bontas
Freie Universität Berlin
Institut für Informatik
Takustr. 9, 14195 Berlin, Germany
paslaru@inf.fu-berlin.de

David Schlangen
Universität Potsdam
Institut für Linguistik
P.O. Box 601553, 14415 Potsdam, Germany
das@ling.uni-potsdam.de

Sonja Niepage
Institut für Pathologie Charitè
Rudolf-Virchow-Haus
Schumannstr. 20-21, 10117 Berlin, Germany
sonja.niepage@charite.de

Abstract

Although a wide range of medical ontologies has already been deployed in the last decade, most of them follow design principles different to those required by Semantic Web applications and consequently can not be directly integrated and reused. We describe a method to construct a Semantic Web-based medical ontology, which is used for the semantic annotation of medical reports, and evaluate the engineering process against a reuse-oriented approach.

1. Introduction

Despite of the variety of medical ontologies developed so far— UMLS[1], SNOMED[2], GeneOntology[3] to name only a few— these ontologies can not be directly integrated into real-world medical information systems. Most of the available ontologies are not formalized in an appropriate representation language to be shared and reused. Additionally, though containing huge amounts of valuable domain knowledge, they are at the same time too comprehensive and task-specific and consequently have to be customized for new application settings [12, 3, 9].

On the other hand, the alternative of building an application ontology from scratch remains a challenging, time-consuming and error-prone task. This is especially true for knowledge-intensive domains such as medicine, since there domain experts are forced to conceptualize large amounts of implicit knowledge *explicitly* and to *re-organize* it in typical ontological categories. For this reasons, knowledge acquisition using natural language processing techniques is often seen as a means to make this tedious process more efficient. Though this method cannot currently be used to automatically generate a domain ontology for a specific purpose, it can be used to *assist* domain experts along the conceptualization process. Besides, this bottom-up oriented approach offers some advantages in cases where the application ontology is to be used for language-related tasks like semantic annotation, since it facilitates the development of "NLP-friendly" ontologies (see Section 3).

In this paper we describe the engineering of a medical ontology—within a certain setting, namely that of semantic annotation of texts—by using NLP-based knowledge acquisition techniques that analyze texts from the target domain.[4] Compared to existing systems [2, 4, 6, 7], these techniques are "knowledge-lean", since they do not require additional *domain* or *linguistic* knowledge or resources, thus reducing the engineering costs accruing from the development of a domain-specific lexicon. As a novel feature they make use of the WWW as a text collection against which the domain texts are compared during analysis; this makes them easy to employ for arbitrary domains even if no linguistic expertise is available.[5]

The rest of this paper if organized as follows: Section 2 describes the NLP tools developed to aid ontology build-

1 http://www.nlm.nih.gov/research/umls
2 http://www.snomed.org
3 http://www.geneontology.org

4 This work has been partially supported by the EU Network of Excellence "KnowledgeWeb" (FP6-507482) and by the project "A Semantic Web for Pathology" which is funded by the DFG (German Research Foundation).

5 We will make the programs publicly available by the time of the conference.

ing. Their use for the development of a medical ontology is described in Section 3. In Section 4 we evaluate the NLP methods used from a technical perspective and analyze the costs and benefits of our engineering approach against a reuse-oriented experiment, aiming at developing a similar medical ontology on the basis of UMLS. We outline future work in Section 5.

2. The OntoSeed Suite

The OntoSeed suite consists of a number of programs that, given a collection of documents from a certain domain, produce various statistical reports (as described below), with the aim to provide guidance for the ontology engineer on which concepts are important in this domain, and implicitly on potential semantic relationships among concepts. More specifically, it compiles five lists for each given collection of texts: i) a list of nouns (or noun sequences in English texts) occurring in the collection, ranked by their "termhood" (i.e. their relevance for the text domain); ii) nouns grouped by common prefixes and iii) suffixes, thereby automatically detecting compound nouns; iv) adjectives together with all modified nouns; and v) nouns with all modifying adjectives.

In the first processing step we determine the part of speech of each word token in the collection. This enables us to extract a list of all occurring nouns (or, for English, noun sequences, i.e., compound nouns; German compound nouns are, as is well known, written as one orthographic word). The "termhood" of each noun is determined by the usual *inverted document frequency* measure (tf.idf) [8], as shown in the formula below—with the added twist, however, of using a WWW-search engine to determine the document frequency in the comparison corpus.[6]

In the formula, $tf(w)$ stands for the frequency of word w in our collection of texts; $wf(w)$ is the number of documents in the corpus used for comparison that contain w, i.e., the number of hits for query w reported by the search engine used (in our experiments, Google and Yahoo). N is the size of the collection, determined in an indirect way (as the search engines used do not report the number of pages indexed) by making a query for a high-frequency word such as "the" for English or "der" for German.

$$weight(w) = (1 + \log tf(w)) * (\log \frac{N}{wf(w)})$$

In the next step, nouns are clustered, to find common pre- and suffixes. We use a linguistically naïve (since it only looks at strings and ignores morphology), but efficient method for grouping together compound nouns by common parts. This step is performed in two stages: first, preliminary clusters are formed based on a pre- or suffix similarity of three or more letters (i.e., "lung" and "lung pathology" would be grouped into one cluster, but also "prerogative" and "prevention"). These preliminary clusters are then clustered again using a hierarchical clustering algorithm [8], which determines clusters based on maximized pre- or suffix length. The compilation of the adjective lists from the tokenized and POS-tagged text collection is straightforward. We now turn to the usage of OntoSeed to engineer a medical ontology for the domain of *lung pathology*.

3. Engineering the Medical Ontology

The project "A Semantic Web for Pathology" investigates the use of ontologies in an information system for image and text data in the medical domain. The underlying ontology is used for the semantic annotation of medical data (i.e. medical reports in text form). The ontology should cover both domain knowledge (i.e. the domain of "lung pathology"), and application-specific data, like the structure and content of medical reports, typical for the health-care institution involved in the project. Additionally, using the ontology for semantic annotation requires a maximal coverage of the vocabulary used by domain experts in medical reports.[7]

Following current ontology engineering methodologies [5], the ontology was developed in the following stages:

- Analysis of the domain: During intensive collaboration with domain experts (pathologists) we identified the key sub-domains in medicine, which should be covered by the target ontology. The intended use of the ontology for semantic annotation requires a "linguistics-friendly" ontology. Therefore the path from lexical items (e.g. words) to ontology concepts should be as simple as possible, which means that ontology concepts should be denominated in a predictable linguistic form [1]. Additionally the concepts should be labelled using the same natural language as the document to be annotated (i.e. for our application, German). Knowledge sources, potentially relevant for our domain are available medical ontologies such as UMLS and medical reports in textual form.
- Conceptualization of the domain knowledge: A collection of pathology reports was analyzed by the OntoSeed tools and the generated results were used to assist the conceptualization process, which was performed manually.
- Implementation, refinement and evaluation: We realized a prototype ontology, represented in OWL, and are currently evaluating it for semantic annotation purposes.

6 Using the Web as a corpus in linguistic analysis has recently become a popular method in computational linguistics; to our knowledge, however, the system presented here is the first to use the Web in this kind of application.

7 Further details about the application setting are described in [9, 14], *inter alia*.

In the following we describe the use of OntoSeed for the ontology conceptualization task.[8] This task is ultimately still performed *manually*; however, compared to a fully manual process, preparing the text information using OntoSeed offers important advantages in tasks like selecting relevant concepts and creating properties such as sub-class relationships. For selecting the relevant concepts, the ontology engineer uses the list of nouns that are ranked according to their domain specificity as described above and selects relevant concept names. Domain-specific and therefore potentially ontology-relevant terms are assigned higher rankings in the noun list (see Section 4 for the evaluation of the ranking function). For example terms like "Tumorzelle/tumor cell" or "Lungengewebe/lung tissue" get assigned a relatively high weight by our analysis methods, which suggests that these terms denote relevant domain concepts that need to be modelled. Terms like "Information/information" are ranked very low, thus being most likely domain-irrelevant.

After simple concept names from the noun list have been identified as being relevant for the ontology scope, the next step is to look up clusters in which these simple domain-relevant concepts occur. The overview of the data afforded by ordering phrases in prefix and suffix clusters can be very useful in deciding how to model complex concepts, since there is no unique way to model them. For example, a noun like "Tumorzelle/tumor cell" can be modelled as a single concept subclass of `Zelle`(cell) or by means of a property like `Zelle locationOf Tumor`. The suffix clustering offers valuable information about sub-classes or types of a certain concept. The prefix clustering can be utilized to identify concept parts or properties. For example concept names such as `Lungengewebe` (lung tissue) or `Lungengefaess`(lung vessel) are placed in the same prefix cluster and identify physical parts of the concept denominated by their common prefix (i.e. `Lunge`(lung)).

The adjective lists give us information about the proportion of all occurrences of a given noun which are in conjunction with a given adjective. This we use to support the ontology engineer's decision to either model the adjective as a property of a concept, or to model the adjective–noun combination as one complex concept. To give an example, the adjective "link/left" occurs only in conjunction with nouns that denote body parts (such as "Lunge/lung"), which indicates that this should be the range of the property. The adjective "gross/large", on the other hand, occurs frequently, and with many kinds of nouns, which indicates that this is a general property.

Further properties are defined by inspecting the conceptualization of relevant compound terms. For example if "Tumorzelle/tumor cell" is to be conceptualized in the ontology as `Zelle locationOf Tumor` the property `locationOf` should also be included to the ontology[9].

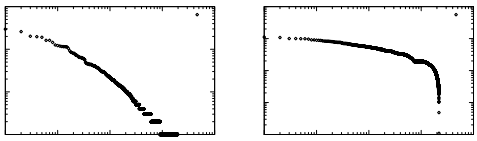

Figure 1. Rank vs. frequency (left) and weight (right); double logarithmically

4. Evaluation

We now first compare the results of our analysis procedure on two different corpora against a naïve baseline assumption and then report an application-based evaluation of the whole suite of tools.

A simple concept of the importance of a term would just treat its position in a frequency list compiled from the corpus as an indication of its termhood. This ranking, however, is of little discriminatory value, since it does not separate frequent *domain-specific* terms from other frequent terms, and moreover, it does not bring any structure to the data: Figure 1 (left) shows a double logarithmic plot of frequency-rank vs. frequency for the LungPath data set; the distribution follows closely the predictions of Zipf's law [8], which roughly says that in a balanced collection of texts there will be a low number of very frequent terms and a high number of very rare terms.

In comparison, after weighing the terms as described above, the distribution looks like Figure 1 (right), again double logarithmically rank (this time: rank in weight-distribution) vs. weight. There is a much higher number of roughly similarly weighted terms, a relatively clear cut-off point, and a lower number of low-weight terms. A closer inspection of the weighed list showed that it distributed the terms from the corpus roughly as desired: the percentage of general terms within each 10% chunk of the list (sorted by weight) changed progressively from 5% in the first chunk (i.e., 95% of the terms in the highest ranked 10% denoted domain-specific terms) to 95% in the last chunk (with the lowest weights). We repeated this process (weighing, and manually classifying terms as *domain-specific* or *general*) with another corpus, a collection of 244 texts (approximately 80,500 word tokens altogether) describing environ-

8 For this purpose we used the LungPath-Corpus [11], consisting of 750 reports of around 300 words each; the preliminary ontology was generated on the basis of a "training-subset" of 400 documents from this corpus.

9 A possible next step in specifying possible ontology properties could be to consider verbs in correlation with noun phrases. Our tool does not yet include this feature, but see discussion below in Section 5.

mental aspects of world countries, and found a similar correlation between weight and "termhood".

In both corpora, however, there was one interesting exception to this trend: a higher than expected number of terms in one 10% chunk in the middle of the weight distribution classified as irrelevant by the experts. These turned out to mostly be misspellings of names for general concepts—a kind of "noise" in the data to which the termhood measure is vulnerable (since in the misspelled form they will be both rare in the analyzed collection as well as the comparison corpus, the Web, pushing them into the middle ground in terms of their weights). While this is not a dramatic problem, we are working on ways of dealing with it in a principled manner.

We now turn to a qualitative evaluation of the usefulness of OntoSeed in a given application setting (as described in Section 3). We compare the effort invested in two semi-automatic ontology engineering experiments which aimed at building the target ontology for the domain of lung pathology, as well as the results achieved by applying these ontologies to semantically annotate medical reports. In a first experiment the ontology was compiled on the basis of UMLS, as the largest medical ontology available. The engineering process was focused on the customization of pre-selected UMLS libraries w.r.t. the application requirements and resulted in an ontology of approximately 1000 concepts modelling the anatomy of the lung and lung diseases [9]. Pathology-specific knowledge was found to not be covered by available ontologies to a satisfactory extent and hence was formalized manually. In the second experiment the ontology was generated by domain experts with the help of the OntoSeed tools as described in Section 3.[10]

The main advantages of the OntoSeed aided experiment compared to the UMLS-based one are the significant cost savings and the increased fitness of use of the generated ontology for the semantic annotation task. From a resource point of view, building the first ontology involved four times as many resources than the second approach (5 person-months for the UMLS-based ontology with 1200 concepts vs. 1.25 person-months for the "text-close" ontology of a similar size). We note that the customization of UMLS required over 45% of the overall effort necessary to build the target ontology in the first experiment.[11] Further 15% of the resources were spent on translating the input representation formalisms to OWL. The reuse-oriented approach gave rise to considerable efforts to evaluate and extend the outcomes: approximately 40% of the total engineering effort were necessary for the refinement of the preliminary ontology. The effort distribution for the second experiment was as follows: 7% of the overall effort was invested in the selection of the relevant concepts. Their taxonomical classification required 25% of the resources, while a significant proportion of 52% was spent on the definition of additional semantic relationships. Due to the high degree of familiarity w.r.t. the resulting ontology, the evaluation and refinement phase in the second experiment was performed straight forward with 5% of the total efforts. The OWL implementation of the conceptual model necessitated the remaining 11%.

In comparison with a fully manual process the major benefit of OntoSeed according to our experiences is the pre-compilation of potential domain-specific terms and semantic relationships. The efforts required for taxonomical classification of the concepts are comparable to building from scratch, because in both cases the domain experts still needed to align the domain-relevant concepts to a predefined upper-level ontology (in our case UMLS' Semantic Network core medical ontology). The selection of domain-relevant terms was accelerated by the use of the termhood measure as described above since this avoids the manual processing of the domain corpus or the complete evaluation of the corpus vocabulary. The efforts necessary to conceptualize the semantical relationships among domain concepts were reduced by the clustering methods employed to suggest potential sub-class and domain-specific relationships. However the OntoSeed approach assumes the availability of domain-narrow text sources and the quality of its results depends on the quality/domain relevance of the corpus.

Besides cost savings the ontology generated in the second experiment has proved to fit better in the application context. The domain ontology is used in several processing stages of the semantic annotation task, all of which can profit from a good coverage (as ensured by building the ontology bottom-up, supported by OntoSeed) and a "linguistics-friendly" specification.[12] One of those naturally is the step of concept lookup, as the ontology defines the vocabulary of the semantic representation. Moreover, having available concept names in predictable linguistic form simplifies matching natural language phrases to concept names.[13] In a second step, the ontology is used to resolve the meaning of compound nouns. E.g., an occurrence of "lung tumor" would in the previous step be mapped to a representation roughly like `tumor REL lung`. The unknown relation `REL` has to be specified by querying the ontology for possible relations (in our ontology, it would be specified to `localizedInBodyPart`). Here we make use of rules formulated by the ontology engineer during the conceptualization process, which might give us preferences for pos-

10 The knowledge-intensive nature and the complexity of the application domain convinced us to not pursue the third possible alternative, building the ontology from scratch.

11 Customization includes getting familiar with, evaluating and extracting relevant parts of the source ontologies.

12 A detailed description of the ontology-based semantic annotation in our project is given in [11].

13 Note that for ontologies like UMLS there is no guarantee that a concept name would be in a particular form, if present at all.

sible relations, so that this resolution can be realized as a *test* rather than a full relation-lookup. A similar process is performed to resolve the meaning of prepositions (e.g. resolving "with" in "mucosa with chronic inflammation" to `localizedInBodyPart`).

We evaluated the fitness of use of the two medical ontologies developed as described above by setting aside a subset (370 texts) of the LungPath corpus and comparing the number of nouns matched to a concept. Using the ontology created by using OntoSeed (on a different subset of the corpus) as compared to the ontology derived from UMLS resulted in a 10 fold increase in the number of nouns that were matched to an ontology concept—very encouraging results indeed, which indicate that our weighting method indeed captures concepts that are important for the whole domain, i.e. that the results generalize to unseen data. However, this evaluation method does of course not tell us how good the recall is w.r.t. all potentially relevant information, i.e., whether we not still miss relevant concepts—this we could only find out using a manually annotated test corpus, which we are planning to do next. But in any case the increase in matches is a clear improvement, since it is guaranteed that all additional matches are true positives.

The results of the evaluation can of course not be generalized to arbitrary settings. Still, due to the knowledge-intensive character of its processes, medicine is considered a representative use case for Semantic Web technologies. Medicine ontologies have already been developed and used in different application settings. Though their modelling principles or ontological commitments have often been subject of research [10, 13], there is no generally accepted methodology for how these knowledge sources could be *efficiently* embedded in real Semantic Web applications. At the same time, the OntoSeed results could be easily understood by domain experts, enabled a rapid conceptualization of the application domain whose quality could be efficiently evaluated by the ontology users.

5. Conclusions and Future Work

We have presented a way in which ontology engineering and natural language processing techniques can work together at the realization of Semantic Web applications. Starting from a typical setting—the semantic annotation of text documents—we introduced a method that can aid ontology engineers and domain experts in the ontology conceptualization process. We evaluated the analysis method itself on two corpora, with good results, and the whole method within a specific application setting, where it resulted in a significant reduction of effort as compared to adaptation of existing resources. As future work, we are investigating to what extent analyzing verbs in domain specific texts can be used to aid ontology building, and ways to extract more taxonomic information from this source (e.g. information about hyponym (is-a) relations, via the use of the copula (x is a y)), while still being as linguistically knowledge-lean as possible. Lastly, we will evaluate the benefits of using "NLP-friendly" ontologies for the semantic annotation task in more detail.

References

[1] J. A. Bateman. The Theoretical Status of Ontologies in Natural Language Processing. KIT-Report 97, TU Berlin, 1992.

[2] P. Buitelaar, D. Olejnik, and M. Sintek. A Protege Plug-In for Ontology Extraction from Text Based on Linguisitc Analysis. In *Proc. of the ESWS-2004*, 2004.

[3] G. Carenini and J. Moore. Using the UMLS semantic network as a basis for constructing a terminological knowledge base: A preliminary report. In *Proceedings of 17th Symposium on Computer Applications in Medical Care*, 1993.

[4] M. Dittenbach, H. Berger, and D. Merll. Improving domain ontologies by mining semantics from text. In *Proc. of the first Asian-Pacific conference on Conceptual modelling*, 2004.

[5] M. Fernández-López and A. Gómez-Pérez. Overview and analysis of methodologies for building ontologies. *Knowledge Engineering Review*, 17(2):129–156, 2002.

[6] U. Hahn and K. Schnattinger. Towards text knowledge engineering. In *Proc. of the AAAI/IAAI*, pages 524–531, 1998.

[7] A. Maedche and S. Staab. Semi-automatic Engineering of Ontologies from Text. In *Proceedings of the Twelfth International Conference on Software Engineering and Knowledge Engineering (SEKE'2000)*, 2000.

[8] C. D. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, Massachusetts, USA, 1999.

[9] E. Paslaru Bontas, S. Tietz, R. Tolksdorf, and T. Schrader. Generation and Management of a Medical Ontology in a Semantic Web Retrieval System. In *CoopIS/DOA/ODBASE (1)*, pages 637–653, 2004.

[10] D.M. Pisanelli, A. Gangemi, and G. Steve. Ontological Analysis of the UMLS Metathesaurus. *JAMIA*, 5:810 – 814, 1998.

[11] D. Schlangen, M. Stede, and E. Paslaru Bontas. Feeding OWL: Extracting and representing the content of pathology reports. In *Proceedings of the NLPXML 2004*, 2004.

[12] S. Schulz and U. Hahn. Medical knowledge reengineering - converting major portions of the umls into a terminological knowledge base. *Int. Journal of Medical Informatics*, 2001.

[13] S. Schulze-Kremer, B. Smith, and A. Kumar. Revising the UMLS Semantic Network. In *Proc. of the Medinfo*, 2004.

[14] R. Tolksdorf and E. Paslaru Bontas. Organizing Knowledge in a Semantic Web for Pathology. In *Proc. of NetObjectDays*, 2004.