Creating ontologies for content representation—the **OntoSeed** suite

Elena Paslaru Bontas[1], David Schlangen[2], Thomas Schrader[3]

[1]Freie Universität Berlin
Institut für Informatik
AG Netzbasierte Informationssysteme
Takustr. 9, 14195 Berlin, Germany
paslaru@inf.fu-berlin.de

[2]Universität Potsdam
Institut für Linguistik
Angewandte Computerlinguistik
P.O. Box 601553, 14415 Potsdam, Germany
das@ling.uni-potsdam.de

[3]Institute for Pathology Charitè
Rudolf-Virchow-Haus
Schumannstr. 20-21
D-10117 Berlin, Germany
thomas.schrader@charite.de

Abstract. Due to the inherent difficulties associated with manual ontology building, knowledge acquisition and reuse are often seen as methods that can make this tedious process easier. In this paper we present an NLP-based method to aid ontology design in a specific setting, namely that of semantic annotation of text. The method uses the World Wide Web in its analysis of the domain-specific documents, eliminating the need for linguistic knowledge and resources, and suggests ways to specify domain ontologies in a "linguistics-friendly" format in order to improve further ontology-based natural language processing tasks such as semantic annotation. We evaluate the method on a corpora in a real-world setting in the medical domain and compare the costs and the benefits of the NLP-based ontology engineering approach against a similar reuse-oriented experiment.

1 Introduction

Ontologies are widely recognized as a key technology to realize the vision of the Semantic Web and Semantic Web applications. In this context, ontology engineering is rapidly becoming a mature discipline which has produced various tools and methodologies for building and managing ontologies. However, even with a clearly defined engineering methodology, building a large ontology remains a challenging, time-consuming and error-prone task, since it forces ontology builders to conceptualize their expert knowledge *explicitly* and to *re-organize* it in typical ontological categories such as concepts, properties and axioms. For this reason, knowledge acquisition and reuse are often seen as ways to make this tedious process more efficient: though both methods cannot *currently* be used to *automatically* generate a domain ontology satisfying a specific set of requirements, they can be used to *guide* or *accelerate* the modeling process.

Natural language processing techniques have proven to be particularly useful for these purposes [3, 8, 6, 13, 18, 24]. However, existing systems are still knowledge or resource intensive: they may not require much prior knowledge about the *domain* that is to be modeled, but they require *linguistic* knowledge or resources. In this paper we present a method to aid ontology building—within a certain setting, namely that of semantic annotation of texts—by using NLP techniques to analyze texts from the target domain. These techniques are comparably “knowledge-lean”, since 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 even if no linguistic expertise is available and reduces the engineering costs since it avoids building an application-specific lexicon.

The techniques not only aid the ontology engineer in deciding which concepts to model, but they also suggest ways to specify the ontology in such a way that it fits ideally into further NLP-based processing steps, e.g. the extraction of information from domain-specific texts. Describing these specification issues and giving an example use case of ontologies thus created is the second aim of this paper.

The remainder of this paper is organized as follows: we motivate our approach and discuss previous work in Section 2. Section 3 gives details about our approach to using NLP to aid ontology design, which is evaluated from a technical and application perspective in Section 4. We close with a discussion of the results and an outline of future work in Section 5.

2 Motivation

2.1 Ontology engineering

Due to the difficulties and costs involved in building an ontology from scratch, ontology engineering methodologies [9] often recommend to rely on available domain-related ontologies or unstructured data like text documents in conjunction with knowledge acquisition techniques, in order to simplify the domain analysis and the conceptualization of the domain knowledge.

In our own experience in a Semantic Web project in the medical domain (see [22, 30] for a longer discussion of this issue, and Section 4.2 below for the project setting), we found that just selecting and extracting relevant sub-ontologies (e.g. from a comprehensive medical ontology like UMLS[1]) was a very time-consuming process. Besides, this approach still resulted in a rather poor domain coverage as determined by the semantic annotation task. The ontology generated in this way could not be involved optimally in NLP-based processes and its acceptance w.r.t. its users was extremely low because of their difficulties in comprehending and evaluating it; this was our motivation to develop the techniques described here.

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

An alternative to reusing available ontologies or related knowledge sources (e.g. classifications, thesauri) is to employ text documents as an input for the conceptualization process. The most basic way to use texts is to extract *terms* from them, i.e. to determine words that denote domain-specific concepts (e.g. "lymphocyte" in a medical text) as opposed to general concepts (e.g. "telephone" in the same text). While this is often seen as a problem that is more or less solved ([7]; see [15] for a review of methods), the methods employed still rely on the presence of linguistic resources (e.g. corpora of non-domain-specific texts, lexicons; our approach differs in this respect, see below), and in any case are only the first step in a text-based analysis: ideally, the goal is to get a collection of terms that is further structured according to semantic relationships between the terms. There are several systems that go in this direction [3, 8, 6, 13, 18, 24], which however still require the availability of linguistic knowledge and resources, and moreover do not seem to work on all kinds of texts.[2] In general, there is a trade-off between the cost of getting or producing these resources and the simplification these methods offer. Hence our aim was a more modest, but at the present state of the art of the Semantic Web and in the given application scenario [22, 30] a more realistic one: to aid the ontology engineer as far as possible, requiring as little additional resources as possible. Before we come to a description of our approach, however, we briefly review the use of ontologies in NLP, and derive some requirements for "NLP-friendly" ontologies. These requirements are crucial for the development of high quality domain ontologies, which should combine a precise and expressive domain conceptualization with a feasible fitness of use (i.e. in our case, fitness of use in language-related tasks).

2.2 Ontologies in NLP

Ontologies have been used for a long time in many NLP applications, be that machine translation [20], text understanding [14], or dialogue systems (some recent examples are [12, 29]), and are of course central to information-extraction or Semantic Web-related NLP applications [2].

Despite all differences in purpose, a common requirement for an ontology to be considered "linguistics-friendly" (or "NLP-friendly") is that the path from lexical items (e.g. words) to ontology concepts should be as simple as possible.[3] On a more technical level, this requires that access to ontology concepts is given in a standardized form—if access is via names, then they should be in a predictable linguistic form. To give an example of this *not* being the case, the medical ontology UMLS contains concept names in the form "noun, adjective" (e.g. "Asthma, allergic") as well as "adjective noun" (e.g. "Diaphragmatic pleura"), and also concept names that are full phrases or even clauses (e.g. "Idiopathic fibrosing alveolitis chronic form"). Below we describe a method to avoid

[2] These methods rely on relational information implicitly encoded in the use of verbs; one of the domains we tested our approach is marked by a reduced, "telegram"-like text style with an almost complete absence of verbs.

[3] See [1] for a still relevant discussion of these interface issues.

these problems during the ontology engineering process, by making the engineering team aware of the requirements of NLP applications; we also describe the concrete use of an ontology in the task of semantic annotation of text documents.

3 Using the OntoSeed suite in ontology engineering

This section describes the suite of tools we have developed to aid the design of ontologies used in language-related tasks such as semantic annotation. [4]We begin by giving a high-level description of the NLP-aided ontology engineering process, illustrating this with examples from the medical domain and explain the technical realization of the tools.

3.1 Overview and examples

The OntoSeed suite consists of a number of programs that produce various statistical reports (as described below) given a collection of texts from a certain domain, with the aim to provide guidance for the ontology engineer on which concepts are important in this domain, and on the semantic relationships among these concepts. More specifically, it compiles five lists for each given collection of texts, as follows:

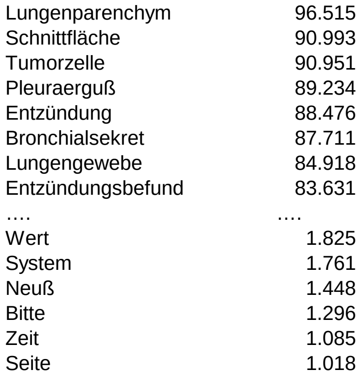
1. a list of nouns (or noun sequences in English texts; we will only write "noun" in the following) occurring in the collection, ranked by their "termhood" (i.e. their relevance for the text domain; see below);
2. nouns grouped by common prefixes and
3. suffixes, thereby automatically detecting compound nouns; and
4. adjectives together with all nouns they modify; and
5. nouns with all adjectives that modify them.

Figures 1 to 3 show excerpts of these files for a collection of German texts from the medical domain of lung pathology (the LungPath-Corpus (see [25]), consisting of 750 reports of around 300 words each; during ontology construction we used a "training-subset" of 400 documents).

As illustrated in Figure 1, terms like "Tumorzelle/tumor cell" or "Lungengewebe/lung tissue" get assigned a relatively high weight by our analysis methods (the highest weight is 112.666), which suggests that these terms denote relevant domain concepts that need to be modeled. Terms related to domain-independent concepts (e.g. terms like "System/system" or "Zeit/time" in Figure 1) tend to be ranked with significantly lower value. Having made the decision to model them, we then look up clusters in which these terms occur, as shown in Figure 2. 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 general, established way to model them. For example, a noun

[4] The OntoSeed tools are available at http://nbi.inf.fu-berlin.de/research/swpatho/ontoseed.html

phrase like “Tumorzelle/tumor cell” can be modeled as a single concept subclass of `Zelle` (cell), while in other settings it can be advantageous to introduce a property like `Zelle infectedBy Tumor` . The suffix clustering offers valuable information about subclasses or types of a certain concept (in our example in Figure 2 several types of cells). The prefix clustering can be utilized to identify concept parts or properties (e.g. in Figure 2 `Lungengewebe` (lung tissue) or `Lungengefaess` (lung vessel) as parts of the `Lunge` (lung)).



Lungenparenchym	96.515
Schnittfläche	90.993
Tumorzelle	90.951
Pleuraerguß	89.234
Entzündung	88.476
Bronchialsekret	87.711
Lungengewebe	84.918
Entzündungsbefund	83.631
....	....
Wert	1.825
System	1.761
Neuß	1.448
Bitte	1.296
Zeit	1.085
Seite	1.018

Fig. 1. Excerpt of the weighted term list (step 1)

Finally, we look at ways in which the relevant terms are modified by adjectives in the texts, by inspecting the lists shown in Figure 3. These lists give us information that can be used in making a decision for one of two ways of modeling the meaning of modifiers: as properties of a concept (e.g. “gross/large” as in “grosse Tumorzelle/large tumor cell”), or as part of a single concept (e.g. “link/left” in `linke Lunge` (left lung)). The decision for either of the modeling alternatives cannot be made automatically, since it depends strongly on the context of the application. However, analyzing a text corpus can support the decision process: modifiers which occur mostly together with particular noun phrases or categories of concepts, respectively, could be candidates for the single concept variant, while those used with a broad range of nouns should usually be modeled as a property. As Figure 3 shows, in our corpus the noun “Tumorzelle/tumor cell” occurs 92 times, 4 times modified with “gross/large” (i.e. approximately 4% of all modifiers). The modifier, on the other hand, occurs 129 times, so the co-occurrences of the two terms are 3% of all its occurrences, which indicates that “gross/large” is a property that is ascribed to many different concepts in the corpus. In contrast, the modifier “link/left” (the normalized form of “links/left”)

B-Zellen

- Carcinom-Zellen
- Schleimhautlamellen
- Plasmazellen
- Epitheloidzellen
- Rundzellen
- Alveolardeckzellen
- Epithelzellen
- Plattenepithelzellen
- Karzinomzellen
- Schaumzellen
- Riesenzellen
- **Tumorzellen**
- Alveolarzellen
- Zylinderzellen
- Becherzellen
- Herzfehlerzellen
- Bindegewebszellen
- Entzündungszellen
- Pilzzellen

Lunge

- Lungen-PE
- Lungenabszeß
- Lungenarterienembolie
- Lungenbereich
- Lungenbezug
- Lungenbiopsat
- Lungenblutung
- Lungenembolie
- Lungenemphysem
- Lungenerkrankung
- Lungenfibrose
- Lungengefäße
- **Lungengewebe**
- Lungengewebsareal
- Lungengewebsprobe
- Lungengewebsstücke
- Lungeninfarkt
- Lungenkarzinom
- Lungenlappen

Fig. 2. Excerpt of the prefix (left, step 2) and suffix lists (right, step 3)

seems to be specific in the corpus to concepts denoting body organs like `Lunge` (lung) and its parts.[5]

To summarize, the classifications of the noun phrases and their modifiers are used as input to the conceptualization phase of the ontology building process, which is ultimately still performed *manually* (Figure 4). Nevertheless, compared to a fully manual process, preparing the text information in the mentioned form offers important advantages in the following ontology engineering sub-tasks:

- selecting relevant concepts: the ontology engineer uses the list of nouns that are ranked according to their domain specificity as described above and selects relevant concepts and relevant concept names. Domain-specific and therefore potentially ontology-relevant terms are assigned higher rankings in the noun list (see Section 4.1 for the evaluation of the ranking function). First simple concept names from the noun list are identified as being relevant for the ontology scope. Then the ontology engineer uses the prefix and suffix clusters to decide which compound concept names should be as well included to the target ontology.
- creating taxonomy: suffix clusters can be used to identify potential subclasses.
- creating properties/relationship: the ontology engineer uses the modifier classification and the generated taxonomy to decide about relevant properties

[5] 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.

Tumorzelle:		92			
	beschrieben	1	1%	10	10%
	einzeln	1	1%	60	1%
	epithelialer	1	1%	1	100%
	gelegen	1	1%	16	6%
	gross	4	4%	129	3%
	klein	1	1%	88	1%
	mittelgross	1	1%	6	16%
	pas-positive	1	1%	6	16%
	spindeligen	2	2%	2	100%
	vergroessert	1	1%	9	11%
	zahlreich	1	1%	47	2%

gross:		
	Absetzungsrand	1
	Abtragungsfläche	1
	Biopsate	1
	Bronchus	2
	Lungengewebsprobe	3
	Lungenlappen	3
	Lungenteilresektat	1
	Lungenunterlappen	5
	Lymphknoten	1
	Nekroseherde	13
	Oberlappenresektat	1
	Ossifikationen	1
	PE	1
	Pleuraerguß	4
	Raumforderung	1
	Rippe	15
	Rundherd	1
	Stelle	5
	Tumor	1
	Tumorknoten	10
	Tumorzelle	4
	Vene	4

Lunge:		85			
	link	9	10%	53	16%
	recht	7	8%	66	10%
	tumorferne	2	2%	2	100%

link:		
	Bronchus	7
	Hauptbronchus	6
	Lunge	9
	Lungenlappen	1
	Lungenoberlappens	1
	Lungenunterlappen	4
	Mittellappen	2
	Oberlappen	9
	Oberlappenbronchus	3
	Seite	1
	Thoraxseite	3
	Unterlappen	4
	Unterlappenbronchus	2
	Unterlappensegment	1
	Unterschenkels	1

Fig. 3. Excerpt of modifier list (steps 4 and 5)

(denoted by adjectives) and about the taxonomy level the corresponding property could be defined. For example in Figure 3 most of the concepts modified by "link/left" are subsumed by `RespiratorySystem` —therefore if the ontology engineer decides to define a property corresponding to this adjective, this property will be assigned the domain `RespiratorySystem` . However since "link/left" occurs in the corpus mostly in correlation with "Lunge/lung" an alternative conceptualization is to introduce the concept `LinkeLunge` (left lung) as a subclass of `Lunge` (lung). Further relationships are induced by the decision to conceptualize relevant compound nouns as two or more related concepts in the ontology. For example if "Tumorzelle/tumor cell" is to be conceptualized in the ontology as `Zelle locationOf Tumor` the relationship `locationOf` should also be included to the ontology. Relationships

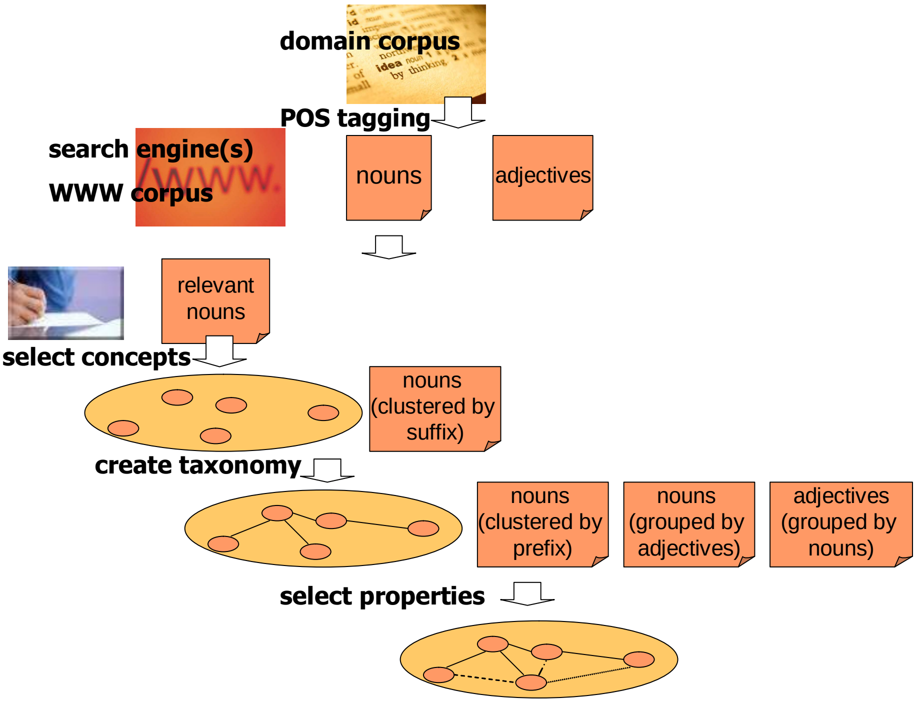


Fig. 4. The OntoSeed process

between concepts (e.g. `locationOf`) are not suggested explicitly; however on the basis of taxonomy which was specified in the previous step OntoSeed is able to identify clusters of compound terms implying a similar relational semantics. For example given the fact that **`Lunge`** (lung) and **`Herz`** (heart) are both subsumed by **`BodyPart`** , the system suggests that the relationship correlating **`Lunge`** (lung) and **`Infarkt`** (attack) in the compound noun "Lungeninfarkt/lung attack" is the same as the one in the case of the compound "Herzinfarkt/heart attack", thus simplifying this conceptualization step even when no linguistic knowledge w.r.t. verbs is available.

3.2 OntoSeed and NLP-friendly ontologies

It is well accepted that NLP-driven knowledge acquisition on the basis of domain-specific text corpora is a useful approach in aiding ontology building [3, 8, 6, 13, 18, 24]. On the other hand, if the resulting ontology is targeted to language-related tasks such as semantic annotation, these tasks can be performed more efficiently by means of an ontology which is built in a "linguistics-friendly" manner. On the basis of our previous experiences in applying ontologies to medical information systems [30, 22] we identified the following set of operations which can be useful in this context and therefore should be taken into account while conceptualizing the ontology:

- logging modeling decisions: the relationship between extracted terms (resulting from the knowledge acquisition process) and the final modeled concepts should be recorded. For example the term `Klatskin tumor` will be probably modeled as a single concept, while `lung tumor` might be formalized as `tumor hasLocation lung` . These decisions should be encoded in a predefined form for subsequent NLP tasks, so that the lexicon that has to be built for these tasks knows about potential compound noun suffixes.
- naming conventions for ontology primitives: since semantic annotation requires matching text to concept names, it is necessary that the concept names are specified in a uniform, predictable manner. [6] Typically concept names are concatenated expressions—where the first letter of every new word is capitalized— or lists of words separated by delimiters (e.g. `KlatskinTumor` or `Klatskin_Tumor`). Furthermore it is often recommended to denominate relationships in terms of verbs (e.g. `diagnosedBy` , `part_of`) and attributes / properties in terms of adjectives (e.g `left`). If the names become more complex, they should be stored in a format that is easily reproducible, and allows for variations. E.g., should there be a need to have a concept name that contains modifiers ("untypical, outsized lung tumor with heavy side sequences"), the name should be stored in a format where the order of modifiers is predictable (e.g. sorted alphabetically), and the modification is disambiguated (`((lung tumor (with ((side sequences), heavy))), (untypical, outsized))`). NLP-tools (chunk parsers) can help the ontology designer to create these normalized names in addition to the human-readable ones.

We now turn to a description of the technical details of OntoSeed.

3.3 Technical details

In the first processing step, the only kind of linguistic analysis proper that we employ is performed: determining the part of speech (e.g., "noun", "adjective", etc.) of each word token in the collection. Reliable systems for performing this task are readily available; we use the TreeTagger [26] developed at IMS in Stuttgart, Germany,[7] but other systems could be used as well.

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), as shown in the formula below—with the added twist, however, of using a WWW-search engine to

[6] This requirement, for example, is *not* fulfilled in UMLS and other medical ontologies.

[7] Freely available for academic research purposes from `http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTagger.html`.

determine the document frequency in the comparison corpus.[8,9] 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, i.e., the number of hits for query w reported by the search engine used— in our experiments, both `www.google.com` (through the API made available by Google inc.) and `www.yahoo.com`. 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.[10]

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

Sorting by this weight results in lists like those shown partially in Figure 1 above; a quantitative description of the effect of this weighing procedure is given in Section 4.1.

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 [19], which determines clusters based on maximized pre- or suffix length (see Figure 2 above). The accuracy of the suffix clustering procedure is anew improved by using the Web to eliminate suffixes that do not denominate concepts in the real world, but are simply common endings of the clustered nouns (such as the ending "ight" in "light" or "night" in English or the German ending "tion" in "Reaktion/reaction", "Infektion/infection").

The compilation of the adjective lists (Figure 3) from the tokenized and POS-tagged text collection is straightforward and need not be explained here.

4 Evaluation

This section is dedicated to the evaluation of our approach from a technical and an application-oriented perspective. We first compare the results of our analysis procedure on two different corpora against a naïve baseline assumption (Section

[8] See [19] for a textbook description of the family of tf.idf measures.

[9] Using the Web as a corpus in linguistic analysis has become a hot topic recently in computational linguistics (see e.g. a current special issue of *Computational Linguistics* [16]); to our knowledge, the system presented here is the first to use the web in this kind of application.

[10] This of course is just an approximation, and also the hits reported for normal queries get progressively less exact the more frequent a term is; for our purposes, this is precise enough, since for "web-frequent" terms (where wf ranges from 10^3 to 10^6) rough approximations already have the desired effect of pushing the weight down.

4.1). The whole suite of tools is then evaluated within a real-world application setting in the medical domain. For this purpose we will compare two engineering experiments aiming at developing the same ontology—a OntoSeed-aided engineering approach and reuse-oriented one—in terms of costs and suitability of the outcomes in the target application context (Section 4.2).

4.1 Technical evaluation

For the technical evaluation of our methods we examined the weighing function described above and the results of the prefix and suffix clustering against human expertise.

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 5 (left) shows a doubly logarithmic plot of frequency-rank vs. frequency for the LungPath data set; the distribution follows closely the predictions of Zipf's law [31], 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.

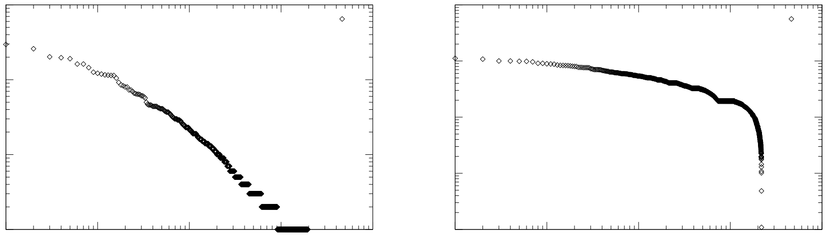


Fig. 5. Rank (x-axis) vs. frequency (left), and rank vs. weight (right); doubly logarithmically

In comparison, after weighing the terms as described above, the distribution looks like Figure 5 (right), again doubly 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 80500 word tokens altogether) describing environmental aspects of world countries, and found a similar correlation between weight and "termhood" (the results for both corpora are shown in Figure 6).

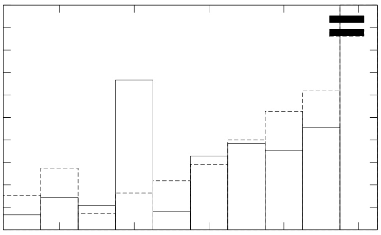


Fig. 6. The ratio of general terms per 10% chunk of weighted term list (highest weight to the left); LungPath corpus (dashed lines) and travel corpus (solid lines)

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 which were 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.

Further on, the comparison of the clusters generated as described in Section 3.3 with the results of the human classification revealed an average percentage of approximately 14% of irrelevant suffix/prefix clusters — a satisfactory result given the linguistically naïve algorithms employed.

We now turn to a qualitative evaluation of the usefulness of OntoSeed within a real-world Semantic Web application we are developing for the medical domain.

4.2 Application-based evaluation

In order to evaluate the costs and the benefits of the OntoSeed approach, we examined two subsequent semi-automatic ontology engineering experiments which aimed at building an ontology for a Semantic Web application in the domain of lung pathology [30, 22]. The application operates upon an archive of medical reports (the LungPath-Corpus mentioned above) consisting of both textual and image-based data, which are semantically annotated in order to transform them into a valuable resource for diagnosis and teaching, which can be searched in a fast, *content-based* manner [22, 30]. The semantic annotation of the data is realized by linguistically extracting semantic information from medical reports and lists of keywords associated with each of the digital images (both reports and keyword lists are available in textual form). The search is content-based in that it can make use of semantic relationships between search concepts and those occurring in the text. In the same time the medical information system can provide quality assurance mechanisms on the basis of the semantic annotations of the patient records. The annotated patient records are analyzed on-the-fly by the quality assurance component, and potential inconsistencies w.r.t. the background domain ontology are spotted.

Extracting semantic information from the medical text data is realized automatically using LUPUS—Lung Pathology System [25]. LUPUS consists of a 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 result of the linguistic analysis is a (partial) semantic representation of the content of the textual data in form of an OWL semantic network of instances of concepts and properties from the domain ontology. This ontology is used in three processing stages in LUPUS, 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 (as described above). The most obvious step where NLP and ontology interface is concept lookup: the ontology defines the vocabulary of the semantic representation. Since LUPUS cannot "know" whether a phrase encountered (e.g. "anthrakotischer Lymphknoten/anthracotic lymph node") is modelled as a simple or complex concept (i.e., as a concept `AnthrakotischerLymphknoten` or as a concept `Lymphknoten` having the property `anthrakotisch`) it has to first try the "longest match". For this to work, the system has to be able to construct a form that would be the one contained in the ontology. To stay with this example, an inflected occurrence of these terms, e.g. in "die Form des anthrakotischen Lymphknotens" ("the form of the anthracotic lymph node"), would have to be mapped to a canonical form, which then can be looked up. As mentioned above, in ontologies like UMLS there is no guarantee that a concept name would be in a particular form, if present at all. In a second step, the ontology is used to resolve the meaning of compound nouns and prepositions [25].

During this project we examined two alternatives for the semi-automatic generation of an ontology for lung pathology which suits the application functionality mentioned above. The two experiments were similar in terms of engineering team (and of course application context). In the first one 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 1200 concepts modeling the anatomy of the lung and lung diseases [22, 21]. 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 with the help of the OntoSeed tools as described in Section 3.1.[11]

We compared the efforts invested in the corresponding engineering processes and analyzed the fitness of use of the resulting ontologies, in our case the results these ontologies achieved in semantic annotation tasks. The main advantages of the OntoSeed-aided experiment compared to the UMLS-based one are the significant cost savings in conjunction with the improved fitness of use of the generated ontology.

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 [12]required over 45% of the overall effort necessary to build the target ontology in the first experiment. 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 necessitated the remaining 11%.

In comparison with a fully manual process the major benefit of OntoSeed according to our experiences would be the pre-compilation of potential domain-specific terms and semantic relationships. The efforts invested in the 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 pre-defined upper-level ontology (in our case the Semantic Network core medical ontology from UMLS). The selection of domain-relevant terms was accelerated by the usage of the termhood measure as described above since this avoids the manual processing of the entire 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 subClass 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.

[11] 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.

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

In order to evaluate the quality of the outcomes (i.e. the ontologies resulted from the experiments mentioned above) we compared their usability within the LUPUS system 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, a task which is currently performed. In a preliminary evaluation, domain experts selected the most significant (w.r.t their information content) concepts from an arbitrary set of 50 patient reports. These concepts are most likely to be used as search terms in the envisioned system because of their high domain relevance (as assigned by human experts). The ontology derived from UMLS contained 40% of these concepts. However, only 8% of them were directly found in the ontology,[13] while the usage of the remaining 32% in the automatic annotation task was practically impossible because of the arbitrary concept terminology used in UMLS. As underlined before UMLS contains concept names in various forms ("noun, adjective", "adjective noun", full phrases—to name only a few). In comparison, the OntoSeed-generated ontology was able to deliver 80% of the selected concepts with an overall rate of 61% directly extracted concepts. In contrast to the UMLS-oriented case, the 19% of the remaining, indirectly recognized concepts could be de facto used in automatic annotation tasks, due to the NLP-friendly nature of the ontology. In the second ontology the concepts were denominated in an homogeneous way and critical modeling decisions were available in a machine-processable format.

The results of the evaluation can of course not be entirely 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 [17]. Medicine ontologies have already been developed and used in different application settings: GeneOntology [5], NCI-Ontology [11], LinKBase [4] and finally UMLS. Though their modeling principles or ontological commitments have often been subject of research [28, 23, 27, 10], 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. Though OntoSeed was evaluated in a particular application setting,

[13] Directly extracted concepts are the result of simple string matching on concept names or their synonyms. The indirect extraction procedure assumes that a specific concept available in the text corpus is formalized "indirect" in the ontology i.e. as a set of concepts and semantical relationships; see Section 3.

that of semantically annotating domain-narrow texts using NLP techniques, we strongly believe that the tools and the underlying approach are applicable to various domains and domain specific corpora with similar results. This assumption was in fact confirmed by the technical evaluation of the tools on a second English corpus from the domain of tourism.

5 Conclusions and Future Work

In this paper we presented methods to aid the ontology building process. 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. Additionally, the method suggests guidelines for building "linguistics-friendly" ontologies, which perform better in ontology-based NLP tasks like semantic annotation.

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 hypnoym (is-a) relations, via the use of the copula (x is a y)), while still being as linguistically knowledge-lean as possible. Second, we are currently implementing a graphical user interface to simplify the usage of the presented tools in ontology engineering processes and in the same time to extend the automatic support provided by the OntoSeed approach. Lastly we will complete the evaluation of the LUPUS system and the benefits of using "NLP-friendly" ontologies for the semantic annotation task in more detail.

Acknowledgements

This work has been partially supported by the EU Network of Excellence "KnowledgeWeb" (FP6-507482). The project "A Semantic Web for Pathology" is funded by the DFG (German Research Foundation). We are also grateful to Google Inc. for making available their API to the public. Thanks to Manfred Stede for valuable comments on a draft of this paper.

References

1. J. A. Bateman. The Theoretical Status of Ontologies in Natural Language Processing. KIT-Report 97, Technische Universität Berlin, May 1992.
2. K. Bontcheva, H. Cunnigham, V. Tablan, D. Maynard, and H. Saggion. Developing Reusable and Robust Language Processing Components for Information Systems using GATE. In *Proceedings of the 3rd International Workshop on Natural Language and Information Systems NLIS02*. IEEE Computer Society Press, 2002.

3. P. Buitelaar, D. Olejnik, and M. Sintek. A Protege Plug-In for Ontology Extraction from Text Based on Linguisitc Analysis. In *Proceedings of the European Semantic Web Symposium ESWS-2004*, 2004.
4. W. Ceusters, B. Smith, and J. Flanagan. Ontology and Medical Terminology: Why Description Logics are Not Enough. In *Proc. Towards An Electronic Patient Record, TEPR2003*, 2003.
5. The Gene Ontology Consortium. Gene Ontology: Tool for the Unification of Biology. *Nature Genetics*, 25:25–30, 2000.
6. M. Dittenbach, H. Berger, and D. Merll. Improving Domain Ontologies by Mining Semantics from Text. In *Proceedings of the first Asian-Pacific conference on Conceptual modelling*, pages 91–100. Australian Computer Society, Inc., 2004.
7. P. Drouin. Detection of Domain Specific Terminology Using Corpora Comparison. In *Proceedings of the International Language Resources Conference LREC04*, Lisbon, Portugal, May 2004.
8. D. Faure and Poibeau T. First Experiments of Using Semantic Knowledge Learned by ASIUM for Information Extraction Task Using INTEX. In *Ontology Learning ECAI-2000 Workshop*, 2000.
9. M. Fernández-López and A. Gómez-Pérez. Overview and Analysis of Methodologies for Building Ontologies. *Knowledge Engineering Review*, 17(2):129–156, 2002.
10. A. Gangemi, D. M. Pisanelli, and G. Steve. An Overview of the ONIONS Project: Applying Ontologies to the Integration of Medical Terminologies. *Data Knowledge Engineering*, 31(2):183–220, 1999.
11. J. Golbeck, G. Fragoso, F. Hartel, J. Hendler, B. Parsia, and J. Oberthaler. The National Cancer Institute's Thesaurus and Ontology. *Journal of Web Semantics*, 1(1), 2003.
12. I. Gurevych, R. Porzel, E. Slinko, N. Pfleger, J. Alexandersson, and S. Merten. Less is More: Using a Single Knowledge Representation in Dialogue Systems. In *Proceedings of the HLT-NAACL Workshop on Text Meaning*, 2003.
13. U. Hahn and K. Schnattinger. Towards Text Knowledge Engineering. In *Proceedings of the AAAI/IAAI*, pages 524–531, 1998.
14. J. R. Hobbs, W. Croft, T. Davies, D. Edwards, and K. Laws. Commonsense metaphysics and lexical semantics. *Compuational Linguistics*, 13(3–4), 1987.
15. K. Kageura and B. Umino. Methods of Automatic Term Recognition. *Terminology*, 3(2):259–289, 1996.
16. A. Kilgarriff and G. Grefenstette. Introduction to the Special Issue on the Web as Corpus. *Computational Linguistics*, 29(3):333–348, September 2003.
17. KnowledgeWeb European Project. Prototypical Business Use Cases (Deliverable D1.1.2 KnoweldgeWeb FP6-507482), 2004.
18. A. Maedche and S. Staab. Semi-automatic Engineering of Ontologies from Text. In *Proceedings of the 12th International Conference on Software Engineering and Knowledge Engineering SEKE2000*, 2000.
19. C. D. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, Massachusetts, USA, 1999.
20. S. Nirenburg and V. Raskin. The Subworld Concept Lexicon and the Lexicon Management System. *Computational Linguistics*, 13(3–4), 1987.
21. E. Paslaru Bontas, M. Mochol, and R. Tolksdorf. Case Studies in Ontology Reuse. In *Proceedings of the 5th International Conference on Knowledge Management IKNOW05*, 2005.
22. 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.

23. D.M. Pisanelli, A. Gangemi, and G. Steve. Ontological Analysis of the UMLS Metathesaurus. *JAMIA*, 5:810 – 814, 1998.
24. M. L. Reinberger and P. Spyns. Discovering Knowledge in Texts for the Learning of DOGMA-inspired Ontologies. In *Proceedings of the Workshop Ontology Learning and Population*, ECAI04, pages 19–24, Valencia, Spain, August 2004.
25. D. Schlangen, M. Stede, and E. Paslaru Bontas. Feeding OWL: Extracting and Representing the Content of Pathology Reports. In *Proceedings of the NLPXML Workshop 2004*, 2004.
26. H. Schmid. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, 1994.
27. S. Schulze-Kremer, B. Smith, and A. Kumar. Revising the UMLS Semantic Network. In *Proceedings of the Medinfo 2004*, 2004.
28. B. Smith, J. Williams, and S. Schulze-Kremer. The Ontology of GeneOntology. In *Proceedings of the AMIA*, 2003.
29. M. Stede and D. Schlangen. Information-Seeking Chat: Dialogues Driven by Topic-Structure. In *Proceedings of Catalog (the 8th Workshop on the Semantics and Pragmatics of Dialogue SemDial04)*, pages 117–124, 2004.
30. R. Tolksdorf and E. Paslaru Bontas. Organizing Knowledge in a Semantic Web for Pathology. In *Proceedings of the NetObjectDays Conference*, 2004.
31. G. K. Zipf. *Human Behaviour and the Principle of Least Effort.* Addison-Wesley, Cambridge, MA, USA, 1949.