[FSt05] *Andreas Faatz, Ralf Steinmetz;* **An Evaluation Framework for Ontology Enrichment.** Ontology Learning from Text: Methods, Evaluation an Applications, IOS Press, Amsterdam, Juli 2005, S. unbekannt.

An Evaluation Framework for Ontology Enrichment

Andreas Faatz [a,1] and Ralf Steinmetz [a]

[a] *Darmstadt University of Technology*

Abstract. Ontology enrichment algorithms propose new concepts to given concepts in a domain specific ontology. The paper is dedicated to a quality measurement framework for ontology enrichment algorithms. We assume that the given ontology does not necessarily provide synonyms or several descriptors per concept. Our basic contribution is the generalization of known quality measures for robust enrichment algorithms. The measures reflect relevance and overlap heuristics. These heuristics are the basis for the evaluation of concept propositions retrieved by ontology enrichment algorithms. We will achieve independence from user evaluations and rely on ontologies, text corpora and on the output of the algorithms.

1. Overview

Ontologies provide models which can be used by a broad range of applications such as for example search and retrieval, advanced web services and software agent communications. Ontologies consist of concepts from a knowledge domain. In an ontology, the concepts are interconnected by semantic relations.
There exist several accepted work-flows for ontology engineering, for example with a strong involvement of knowledge representation experts (as in the METHONTOLOGY approach [GFC04]), as collaborative processes of domain experts with a rather loose moderation [HJ02] or with automated support [MS01]. State-of-the-art methodologies of ontology engineering work-flows can be enhanced by so-called ontology enrichment techniques. Ontology enrichment techniques are automatic processes which generate extensions of the ontology and propose these extensions to the ontology engineers. Thus ontology enrichment can be an embedded part of an iterative process of ontology creation, where ontology engineers try to improve a given ontology.
In general the extensions of the ontology can include new concepts to be integrated, new relations to be instantiated between existing concepts and corrections of existing concepts and relations.
Referring to ontology engineering methods, especially Holsapple and Joshi [HJ02] are very explicit on the usage of a given ontology. The agreement upon an initial ontology is the first step in this engineering process. For example the given ontology can consist of a topic hierarchy from a domain catalogue or a table of contents from a domain specific textbook. The assumed situation differs from a completely automated construction

[1]Correspondence to: Andreas Faatz, Ralf Steinmetz, KOM - Multimedia Communications Lab, TU Darmstadt, Merckstrasse 25, 64283 Darmstadt, Germany, email: {afaatz, rst}@kom.tu-darmstadt.de

of the whole target ontology and provides a central assumption of our work: *a good ontology enrichment algorithm must be able to re-establish a given ontology, upon which the authors already have agreed.* Moreover we will consider only situations, where new concept propositions are placed to the most appropriate position in the given ontology. This happens without the prediction of a relation between the proposition and the given concepts. In terms of Holsapple and Joshi [HJ02] the definition of such a relation would be an iterative step due to the ontology engineers (domain experts). This understanding of ontology enrichment seems reasonable in cases where a graphic visualization tool (see for example [GFC04]) allows for simple edition of relations between concepts. Ontology enrichment in consequence leaves authors with the decision to accept or reject a concept proposition and (in case of acceptance) to draw a relation between the proposition and its surrounding (the concepts from the given ontology). A final assumption refers to a 'formal worst-case situation' during the engineering process: every concept might only be described by its name and its relations to other concepts (that means: it is possible, that neither synonyms nor concept definitions are necessarily given).
Previous work was dedicated to ontology enrichment algorithms, which propose concepts according to linguistic regularities in domain text corpora [FS02], [FHS+02]. Along with the concepts, the algorithm suggests a place among the concepts in the ontology, where the new concept matches best.
Our paper compiles the crucial points of measuring the quality of ontology enrichment for algorithms working with lexical items as propositions. Our aim is to classify ideas for a foundation of quality measures. Work on the quality measures is dedicated to two major benefits: several algorithmic approaches become comparable and fine tuning of ontology enrichment approaches itself becomes possible - analogous to other applications of machine learning like search engines or text categorization.
The most common definition of an ontology states that an ontology is a shared conceptualization of a (knowledge) domain [Gr93]. Although this definition concentrates on the notion of conceptualization, a more technical variant clearly describing the interconnections of the domain concepts will be needed. Therefore our paper is organized as follows. In section 2 we refer to a definition with minimal requirements on components of an ontology. In turn these minimal requirements (i.e. named concept hierarchies) will be applied in our relevance and overlap measures. In section 3 we will introduce ontology enrichment. Section 4 will develop overlap measures for ontology enrichment and we will show approaches to relevance aspects. Finally we show example measurements as applications of our framework and end up with a brief conclusion and with future work.

2. Ontology definitions and assumptions

An ontology is a set of concepts ordered by a subconcept relation. Moreover there exists a set of relation names together with a restriction for each one of them: the restriction expresses which subconcepts of which superconcept are allowed at the i-th place of a relational tupel. We refer to Bozsak et al. [BEH+02] for a detailed ontology definition. For our approach on evaluation we need the subconcept part of an ontology definition, which constitutes a concept hierarchy.

Definition 1 (Concept hierarchy) *The concept hierarchy of an ontology is a pair* $\Omega := (B, \leq)$*, where* B *is a finite set and* $\leq$ *is an order relation on* $B \times B$ *(*$\leq \subseteq B \times B$*). We call* *the $b \in B$ concepts and $\leq$ the subconcept relation. Furthermore there exists an abstract root concept $\top$ for all concepts in B: $\exists(\top \in B)\forall(b \in B) : b \leq \top$.*

Definition 1 does not imply that there is at least one natural language string naming every concept. Therefore we also add the following assumption to the definition of an ontology:

Assumption 1 (Concept descriptor) *Every concept $b \in B$ of the ontology from definition 1 has at least one natural language string as a name. We call this string, which can consist of one or several words, the descriptor $d(b)$ of the concept.*

With assumption 1, a huge class of ontology construction and enrichment problems will find a formal basis.

3. Ontology enrichment

In contrast to conceptual clustering techniques, ontology enrichment starts from a given ontology. It is meant to support real-world ontology engineering processes. We introduce ontology enrichment as a group of approaches which generate for a given concept hierarchy Ω additional concepts as propositions, find a place for these propositions in Ω, operate with statistical data about the usage of the descriptors (names of concepts) of the ontology in a text corpus. The background of this decision comes from the ways an ontology is constructed by domain experts. One way is collecting and ordering lists of related terms and defining a concept by such a group. An ontology engineer might also think of a concept fully described by its name. To the extreme we might end up with exactly one name for each concept.
In contrast to our assumption 1 existing approaches do not necessarily work with lexicalized concepts and lexical descriptors for propositions. For example [TRB02] applies description logic to an automatic instantiation of ontologies, [GW02] introduces metaproperties and logic for ontology maintenance - with new concepts to be integrated - and [SM01] applies Formal Concept Analysis to define new concepts for ontology merging. However our formalization and the related approaches from conceptual clustering follow assumption 1. This will provide a basis for relevance and overlap definitions without the need of user evaluations.

3.1. Formalization of ontology enrichment

Before we start our work on formalizing the notion of enrichment quality by defining numeric measures, the corpus-based ontology enrichment problem itself must be explained in a formal way. We define:

Definition 2 (Ontology enrichment, propositions) *Let ξ be a text corpus, that means, a collection of written or spoken text documents, which are processable for natural language analysis. Let $B(\xi)$ a set of words and phrases from ξ. An ontology enrichment algorithm is an algorithm which takes a given ξ and a given concept hierarchy Ω from an ontology as input and produces for each $b \in B$ a set $P(b) \subseteq B(\xi)$ as output. We call $P(b)$ the set of propositions for b.*

For an ontology enrichment algorithm we explicitly define the set of words or phrases, for which the similarity computations or the decisions if or if not it might become a proposition are calculated at all. With the notations from definition 2 we obtain

Definition 3 (Candidate) *A candidate is a word or phrase $x \in B(\xi)$. A candidate detection mechanism defines the set $B(\xi)$.*

The sets of propositions for an ontology enrichment algorithm can be derived by similarity functions [FS02], [FHS+02]. For given Ω and ξ let $s_0 : B \times B \longrightarrow \mathbb{R}_0^+$ be a similarity function mapping each pair of concepts from the ontology to a similarity value. The algorithm in [FHS+02] extends s_0 to a general similarity function

$$s : B(\xi) \times B(\xi) \to \mathbb{R}_0^+ \tag{1}$$

and the sets of propositions $P(b, t)$ for a given $b \in B$ and a real threshold t can thus be defined as

$$P(b, t) := \{x \in B(\xi) | s(b, x) \geq t\} \tag{2}$$

We see from equations (1) and (2) that the main parameters for the quality of an enrichment algorithm in the sense of [FHS+02] stem from the definition of the similarity functions and the variation of the thresholds. If we define quality measures related to relevance and overlap we would expect a higher overlap for lower t, because

$$|P(b, t_1)| \geq |P(b, t_2)| \Leftrightarrow t_1 \geq t_2 \tag{3}$$

t must not be expressed as one absolute numeric value for all ontology enrichment problems, but as a variable depending on the structure of the ontology and on the chosen concept b.

The core ideas of ontology enrichment differ from automatic ontology construction, for instance conceptual clustering techniques ([N99]). Related work on ontology learning evaluation like [MPS02] considers the following situation. For each given concept, for instance a, b and c, there exist several descriptors. In a set-theoretic notation we find the sets of descriptors (see top of figure 1) $D(a) := \{a_1, a_2, a_3, a_4\}$ for A, $D(b) := \{b_1, b_2, b_3, b_4\} \cup D(a)$ for b, and $D(c) := \{c_1, c_2, c_3, c_4\} \cup D(b)$ for c and therefore

$$D(a) \subseteq D(b) \subseteq D(c) \tag{4}$$

If $|D(a)|, |D(b)|, |D(c)| \geq 2$, then an evaluation technique may leave out some of the descriptors by chance (in our example f, d and j) and judge about their placement after the ontology enrichment process. This approach is applied in [AL01] with structures to be found in the synset hierarchies of thesauri as for example [WordNet]. For each concept in WordNet there exists a set of synonyms, which does not appear in the superconcepts. All approaches with several descriptors underly assumption 1, but our minimal assumption is the existence single descriptors. If we take an ontology engineering process like the one in [HJ02] into account, especially non experts in formalization and conceptual-

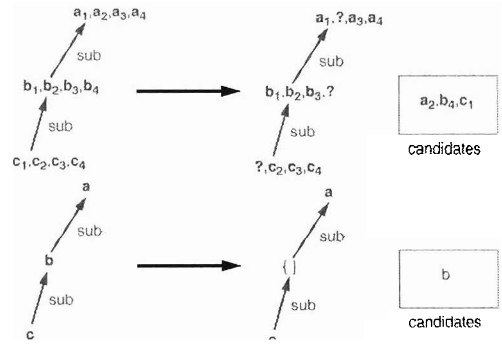


Figure 1. Clustering and enrichment: the initial structures

ization might establish hierarchies of *single* descriptors from scratch - without the additional effort of grouping similar descriptors and placing them into a hierarchy of such groups. This more general situation is depicted in the bottom part of figure 1. We see, that the existence of several descriptors has an impact on evaluation techniques. In line with the process in [HJ02] a non-expert in knowledge representation can also start ontology engineering by grouping descriptors into superconcept-subconcept hierarchies - instead of engineering by formal concept extensions or intensions (compare for example the constructions in [SM01]). The structurally minimal case is the following one: if we just leave out descriptors by chance, we remove all information about a concept (see bottom of figure 1). In the sections on overlap we will suggest a solution to this problem, which does not collide with our ontology definition.

3.2. Ontology enrichment example

Let us finally show two brief and simplified examples of an ontology enrichment technique, which operate with collocation data of a text corpus. Let Ω_{ex} be given with $B_{ex} := \{\top, disease, diarrhoea\}$ and

$$diarrhoea \leq disease \leq \top \qquad (5)$$

Furthermore, consider the following example collocations from a text corpus ξ_{ex}.

disease:heart (1543), AIDS (1063), patients (962), Alzheimer's (773), cancer (544), coronary (429), blood (354), Parkinson's (311), symptoms (304), virus (300)
diarrhoea: symptoms (30), virus (11) vomiting (5) nausea(3) salmonella(1)

The example should be read as "heart" co-occurred with "disease" in ξ_{ex} 1543 times,

"symptoms" co-occurred with "diarrhoea" in ξ_{ex} 30 times et cetera. In this case, co-occurrence refers to being words in the same sentence.
An ontology enrichment approach in the sense of [FHS+02] would identify the set {"symptoms", "virus"} as the co-occurring words of the concept descriptors for both *disease* and *diarrhoea*. The algorithm declares the similarity function s by the degree of similarity to a transformation of the vectors $(30, 11)$ (for *diarrhoea*) and $(304, 300)$ (for *disease*). In other words, a word with many co-occurrences of "symptoms" and "virus" is a potential proposition. The similarity s is derived from a similarity s_0, which would only cover the similarity between *diarrhoea* and *disease*.
In contrast, an ontology enrichment algorithm in the spirit of [AL01] or [Y92] would identify a set S of the co-occurring terms different from the ones in $\{"symptoms", "virus"\}$ and determine a significance interval $[a; b]$. If the number of co-occurrences is inside $[a; b]$, then $s \in S$ becomes a proposition.

4. Measures for automated evaluations

This section consists of three parts. The first part motivates the need for automatic evaluation measures of ontology enrichment algorithms. The second part consists of related work. The remainder of the section is dedicated to relevance and overlap measures with a special focus on the additional assumptions on the enrichment algorithm if we undertake automated evaluations.

4.1. Ontology enrichment processes and parameters

Holsapple and Joshi define an ontology engineering process [HJ02], where the ontology engineers systematically judge about the other ones' conceptual modelling. Each concept which is added to an existing ontology is presented to at least another domain expert, who applies a numerical scale to express the quality of the modelling. In principle, with such an approach one could also judge about ontology enrichment algorithms, but from a development point of view there exists a hurdle: before an ontology enrichment algorithm is applied to support a human-centric ontology engineering process, we need a fine-tuning of the algorithm concerning many open parameters. We now give an overview of these parameters for the case of similarity based ontology enrichment.
First, there are different ways to construct the underlying domain specific text corpus for ontology enrichment. Especially in situations like the one described in [FS02] we meet the question, how World Wide Web (WWW) search results and textual archives should be transformed into a text corpus ξ to derive language regularities from. Generally spoken, corpus construction parameters become complicated with an increasing number of corpus sources. Moreover additional input parameters for similarity-based ontology enrichment depend on the choice of the initial similarity function s_0. The ontology structure can be transformed to s_0 by path-oriented (that means: paths in a graph with edges from B and relations from $\leq$ for a given concept hierarchy $\Omega := \{B, \leq\}$) measures like the ones from [Li03] or [FHS+02]. Another possibility are measures considering information content like the ones in [R99]. At this point we comment, that also the variation of the thresholds t can be highly dependent on the choice of s_0.
Unfortunately, if we extend s_0 to s further parameters occur. These depend on the sim-

ilarity metaphor for the representation of word contexts in the sense of [L99]. Besides that, for general ontology enrichment the question arises, if we should represent syntactic or window-based features from the word contexts [G92].
From the vast amount of parameters influencing ontology enrichment we conclude, that user evaluations for ontology enrichment algorithms are not feasible - the bare amount of evaluation tasks is not acceptable in rapid ontology engineering. Moreover from our point of view there might occur another fundamental problem: the result of an enrichment algorithm might distract the evaluating user from the direction the ontology actually should have evolved. The other way around the evaluator might refuse helpful hints from the algorithmic output - both faults are generally known in information retrieval [BR99]. In ontology enrichment the problem becomes even worse as it is impossible to let a bigger group of users evaluate the results: by a central design requirement, domain ontologies are carved for a particular task and a particular application. Thus, an evaluation must always be aware of the task - thus, actually the ontology cannot be evaluated by a third party different from the ontology engineers. From both circumstances we conclude, that only automatic evaluations of ontology enrichment meet the requirements of algorithmic tuning. Moreover the automatization has to be aware of the particular task specific domain, to which an ontology should evolve.

4.2. Related automated evaluation approaches

For our quality measures we refer to an approach using similarity thresholds instead of conceptual clustering techniques. Once more the background of this decision comes from the way an ontology is constructed by domain experts: instead of collecting and ordering lists of related terms and defining a concept by such a group, an ontology engineer might also think of a concept fully described by its name. This assumption contrasts for example the work of [NVC+04], who generates concept definitions to evaluate ontology learning. If a concept is fully described by its name we might end up with the minimal requirement sated in assumption 1: exactly one name for each concept. This paradigm contrasts to the automated evaluations of [MPS02], where successively exactly one descriptor from a structure like the one at the top of figure 1 is deleted and a test determines, if an ontology enrichment algorithm would propose it at this place again. If we delete a descriptor from a structure like the one in the lower part of figure 1, this technique needs further clarification, because the concept disappeared and we have changed the ontology. Nevertheless we remark, that the task of [MPS02] differs from our general approach: this work aims at augmenting existing concept hierarchies with subconcepts or hyponyms. Our task is more abstract, as assigning certain terms to existing concepts.
Another evaluation paradigm is the cotopy measure [MS02]. This measure is not directly applicable to our problem, because in ontology enrichment the actual relations of the propositions are identified by a human. Using cotopy we would have to identify ontological relations. This would include a prediction of the subconcept relation, that means predicting a generalization/specialization direction to our concept propositions. Nevertheless we point out, that our approach is inspired by the cotopy measure and can be viewed as its adaptation to the more abstract task of ontology enrichment. Our principles of overlap and the general task of fine-tuning ontology learning algorithms are also related to the work of [S04], but we found our definitions on a per-concept basis.
We now start constructing alternatives to the existing approaches.

4.3. *Aspects related to relevance*

To evaluate relevance at all, we have to assume a candidate detection mechanism. Note again the difference between candidates and propositions: for instance, in similarity-based ontology enrichment a candidate is a term, for which a co-occurrence vector is established and a similarity to concepts from the ontology is computed at all. A proposition is a candidate, which under the similarity definition is similar enough to be placed to a given concept. For the following subsections 4.3.1 and 4.3.2 on relevance let a set $B(\xi)$ of candidates be given.
Regarding relevance our central hypothesis is the following one: it is not possible to fully compute relevance if we do not know the conceptual relevance of each candidate. From the moment we consider words or phrases, which do not come from the given ontology, there is no automatic way of judging about their quality: from our point of view quality statements are only allowed for the descriptors we already met with the given concepts. Our attempt in the remaining sections is the following one: we define evaluation measures from observations on how an ontology enrichment algorithms causes an evolution of the ontology. If this evolution is comparable to the construction, which would be the outcome of human ontology engineering, we obtain a high quality of enrichment.

4.3.1. *Percentage of true candidates and relevance*

At a first glance this is more an evaluation measure for the mechanism detecting candidates. Our measure computes the ratio of terms in $B(\xi)$, which are (up to stemming) identic to the concept descriptors from the given ontology. If the ratio is high we obtain a hint on a high quality of the way we defined the set of candidates. High quality in this sense rises the opportunities of the actual enrichment to work well. Note that this evaluation measure contrasts to nearly all directions of conceptual natural language processing (e.g. [G92], [N99], [Y92]), which apply stemming for words to obtain a better statistics on their usage. An example of how different collocation information can turn out for singular and plural forms in the same corpus is listed below. The result was produced by querying the German Wortschatz project [wort], [Q98]. We list the first ten significant collocations together with their total.

disease:heart (1543), AIDS (1063), patients (962), Alzheimer's (773), cancer (544), coronary (429), blood (354), Parkinson's (311), symptoms (304), virus (300)
diseases: AIDS (279), cancer (265), infectious (199), sexually (177), transmitted (163), patients (154), lung (149),immune (140), chronic (113), genetic (112)

We observe remarkable collocation differences for both approaches. We conclude that we are able to produce true candidates via the lexicalization of the existing concept descriptors, but the true candidates may have different corpus characteristics than the original descriptor.
Unfortunately besides candidates resulting from lexical variations there are still other candidates left, which (to one extreme) could all over be equally good candidates or (to the other extreme) all be irrelevant. Moreover if we compare two versions of an ontology enrichment algorithm or two different ontology enrichment algorithms - should refer to the same candidates as long as we enrich the same ontology. From the point of view

that is developed in [FS02] and [FHS+02] this is not self-evident, because the candidate detection mechanism can be a result of the feature selection. In consequence, a candidate detection - generally may purely depend on $\{"symptoms"\}$ in one case and purely on $\{"virus"\}$ in the other case. Instead, for evaluation purposes we should refer to candidate detection by all terms in $\{"virus", "symptoms"\}$.
The second point is that we are forced to a pessimistic judgement of the other candidates. Thus we refer for the remainder of this section to lexical variants $v(\Omega) := \{v_i(b) | b \in B, 0 \leq i \leq n, \}$ (i is the i-th variant and n the total of variants we permit) which we judge as good candidates. The rest of the candidates are considered bad candidates, which become bad propositions if they become elements of $P(b, t_{v(b)})$. A relevance evaluation method for similarity-based ontology enrichment approaches may fix for each $b \in B$ a threshold $t_{v(b)}$ in such a way, that at least one $v_i(b)$ becomes a proposition for b and determine with the notation from equation (2)

$$\frac{|\{x \in P(b, t_{v(b)}) | x \in v_i(b), 0 \leq i \leq N\}|}{|P(b, t_{v(b)})|} \tag{6}$$

This formula may be applied locally, that means for one b, or it may be used to average over the whole ontology.

4.3.2. Discrimination

This measure is another indicator of how well a similarity based ontology enrichment algorithm separates between the candidates. If there are many candidates, but only a few propositions, this *might* be an indicator of relevance. If for a concept c

$$\frac{|\{x\ B(\xi) \mid s(x, c) > t\}|}{|B(\xi)|}$$

is low, then only relatively few of the candidates are actually selected as propositions. From an engineering perspective this can be a desirable situation, instead of overwhelming the ontology engineers with too many propositions. If in turn the ratio is high, a high portion of the candidates is selected and proposed. This can again correspond to a good or bad candidate detection mechanism. Thus a better solution for the determination of algorithms, which are *not* overly sensitive concerning the thresholds, is the maximization of the distance (i.e. minimization of the similarity) between the candidates, which are included as propositions and the candidates, which do not become propositions. In both enrichment examples from section 3.2. this is possible, because the similarity measures are an output of the enrichment algorithms. An example for a discrimination measure for similarity-based approaches is

$$\sum_{b \in B} \min_{x \in B(\xi), x < t(b)} s(x, b) \tag{7}$$

For small values of this measure we obtain a clear discrimination in the sense, that the propositions are relatively robust concerning small variations of the thresholds $t(b)$. Discrimination is a second order goal and inspired by classification approaches like support vector machines [CVB+02]. In our sense discrimination should only be considered with

a given bandwidth of relevance and overlap.
A common additional assumption we met with our relevance definitions is the introduction of a concrete candidate detection mechanism with lexical variants. This means that we need additional knowledge to evaluate an ontology enrichment algorithm. This situation is similar in cases where we would not deal with lexical variants but with synonyms: again for each concept descriptor we would need knowledge about its synonyms. The next section will describe other solutions, which are independent from such additional knowledge. In fact these overlap approaches will apply to ontology enrichment in general.

4.4. Aspects related to overlap

In section 4.3.1 the guiding idea was a re-establishment of the concept hierarchy as a whole. For the remainder of the paper we shift to randomly reducing a given concept hierarchy and checking if it can be re-established by an ontology enrichment algorithm. For a given $\Omega := \{B, \leq\}$ we generalize the idea of [MPS02], but we also extinct existing concepts from a given ontology and collect them in a set C which becomes the candidate set and is independent from additional descriptors from the corpus. The idea of random choice of C can only be applied, if

$$\Omega' := \{B \setminus C, \leq\}, \tag{8}$$

is again a concept hierarchy. Note that for cases with single descriptors a definition of Ω' is neccesary as - assuming the worst-case scenario from the introduction of this paper - removing a descriptor means removing a concept completely. Ω' is a concept hierarchy because by transitivity of $\leq$ we may keep for instance a relation $k \leq m$, if $k \leq l \leq m$ was part of the original concept hierarchy. The restriction relation of the relation $\leq$ still remains transitive. Consequently we only have to claim $\top \notin C$.
Furthermore let $d_\Omega(b_1, b_2)$ denote the shortest relational path along $\leq$ and its inversion $\geq$ between $b_1, b_2 \in B$. Then we define our measure called n-edge-overlap for a given $c \in B$ as the ratio

$$\frac{|P(c) \cap \{b|d_\Omega(b,c) \leq n\}|}{|\{b|d_\Omega(b,c) \leq n\}|} \tag{9}$$

Note that the n-edge-overlap for $n > m$ is not necessarily more or equal than the m-edge-overlap, as the denominator $\{b|d_\Omega(b,c) = n\}$ may grow faster than the corresponding enumerator.

Figure 2 shows an overlap example. From the above ontology two concepts d and a are removed and thus become candidates. We indicate this in the second depicted structure by brightening all subconcept relationships where d and a are involved. The remaining two structures represent enrichment outcomes. Propositions are depicted by drawing a dotted line between a concept and a candidate. In the first case of the two ontology enrichment procedures we obtain a 1-edge overlap and a 2-edge-overlap of 100 per cent for concept b, in the second case below we obtain 50 per cent for the 1-edge-overlap and 100 per cent for the 2-edge overlap for concept b.
Figure 3 depicts a more complicated case, where the random choice of a candidate which leaves the ontology goes along with a change of the ontology structure. The indirect

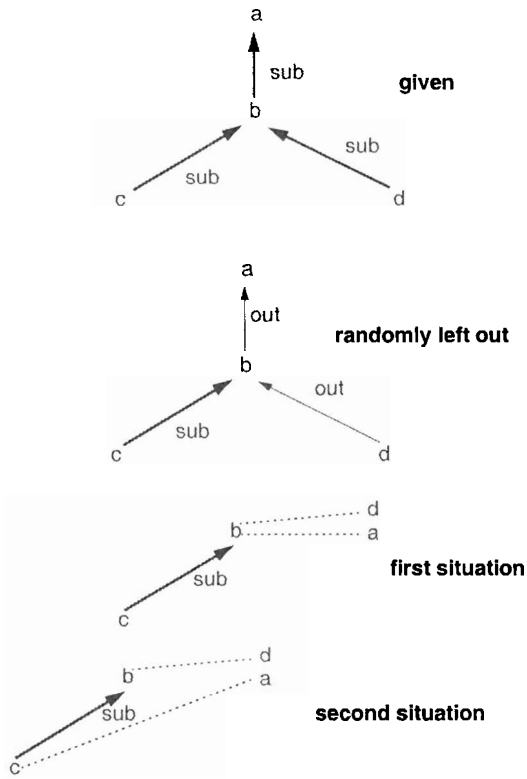


Figure 2. Overlap example

subconcept relations become subconcept relations by transitivity and the candidate b becomes a desirable proposition for all remaining concepts. By variation on $|C|$ we can observe if the enrichment algorithm is able to propose new concepts for more (greater $|C|$) or less (smaller $|C|$) complete ontologies.

Although the measure we presented in equation (9) works under the assumption that the given Ω is correct, this assumption still seems a weaker one than a candidate detection mechanism and a lexical variation in our section on relevance. This motivates the turn of (9) to an enrichment quality measure, which computes the ratio of proposition failures with the aim of measuring overlap. If the n-misclassification

$$\frac{|P(c) \cap \{b|d_\Omega(b,c) > n\}|}{|\{b|d_\Omega(b,c) > n\}|} \tag{10}$$

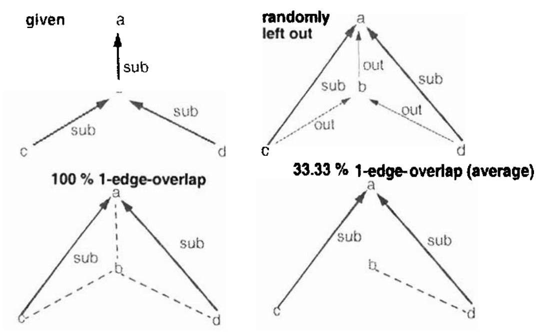


Figure 3. Subconcept structure and overlap measures

is high, we obtain many propositions for a concept c, which are actually out of scope and our relevance decreases. Especially in cases where n is high and would correspond to parts of the ontology, which are less related to the part under actual enrichment consideration, *we speak of outer-ontological enrichment tendency* (in contrast to *inner-ontological enrichment tendency* for low n in (10)). To sum it up, (10) can be understood as alternative to the relevance measures from the subsection before. Finally, we mention that both n-edge overlap and the n-misclassification (10) may be averaged over the concepts of an ontology. Moreover, for n-edge-overlap this average can be weighted with respect to the number of high influences from the concepts which expect many propositions. Analogously, such a weighting scheme may be applied to the averaged proposition failure measure. For average computations, the threshold t can differ for each concept.

5. Framework application example

As we mentioned in the above sections, our evaluation framework was applied in the comparison of different ontology enrichment algorithms. Algorithm 1 and 2 are examples, which we evaluated in a detailed study [F04]. Our notation shows the general procedure of the enrichment algorithms. However, for our evaluation purposes in both cases we use the **DATA**-Input with a reduced concept hierarchy Ω' to be enriched. Ω' consists of concepts $B \setminus C$. The evaluation is based on the concepts, which were left out by random. These are exactly the concepts C. We will sketch the main evaluation results briefly.

Algorithm 1 works in the spirit of the collocation-based example of section 3.2. and can be seen as the core of approaches like the one in [Q98]: once we find a collocation in the text corpus, the respective descriptor becomes a concept proposition. Algorithm 2 refers to the notion of similarity-based ontology enrichment, as we introduced it in 3.2. It translates the structure of the ontology into real similarity values. The vector $\vec{k}$ is responsible for reconstructing this similarity values. Its interpretation is a weighting scheme for features in vectors, which belong to the concepts from the given ontology. These vectors reflect the total numbers of several collocation features for the concept descriptors in a domain specific text corpus. As we may reconstruct different reasonable similarity measures (for the ontology as well as for the vectors), the algorithm can appear in several variants.

We tested both algorithms with small ontology chunks from a medical ontology and a small set of medical web-sites as domain corpus and collocation features in 5-windows (that means: at a maximum distance of five words). We gained significance by taking samples of per-concept misclassification and overlap measures. In particular we found that in terms of overlap variants of algorithm 2 are superior to algorithm 1. The best overlap measures (around 0.32 per concept for the 1-edge and the 2-edge overlap) per concept came along with tuning algorithm 2, whereas these measures were low for algorithm 1 (around 0.1). Other configurations of algorithm 2, which operated dissimilarity measures even showed no enrichment at all (in statistical terms). We also compared n_i-misclassifications in several ranges for n_i with some of the n_i yielding outer-ontological enrichment. The best enrichment in the sense of overlap also produced a misclassification (outer-ontological enrichment), which cannot be distinguished from 0.

Algorithm 1 Naive ontology enrichment

DATA a concept hierarchy $\Omega := (B, \leq)$, a text corpus ξ, an integer δ_W as word distance, candidates C
RESULT concept propositions $P(b)$ for all $b \in B\backslash\{\top\}$

FOR all $b \in B\backslash\{\top\}$: $P(b) \leftarrow \varnothing$.
FOR all $c \in C$:
    IF c collocated at a maximal distance δ_W to the descriptor b in ξ
    THEN $P(b) \leftarrow P(b) \cup c$.

Algorithm 2 Similarity-based ontology enrichment

DATA a concept hierarchy $\Omega := (B, \leq)$, optimal weights $\vec{k}_{opt}$, a text corpus ξ, a vector-valued similarity measure S_v, a real-valued threshold t_b for each concept, candidates C
RESULT concept propositions $P(b)$ for all $b \in B\backslash\{\top\}$
FOR all $b \in B\backslash\{\top\}P(b) \leftarrow \varnothing$
FOR all $c \in C$ $S_c \leftarrow \vec{k}_{opt}S_v(b, c)$:
    IF $S_c > t_b$ **THEN** $P(b) \leftarrow P(b) \cup \{c\}$

---

6. Conclusion and outlook

We presented measures for automatic approaches to ontology enrichment evaluations. The overlap measures exploit the given ontology structures, are independent from addi-

tional assumptions on candidates and can be extended to a proposition failure measure. The latter one reflects the lexical relevance of the total of the propositions per concept. We conclude that the strategy of a random extinct of concepts from an ontology to evaluate enrichment results is easier in the sense of applications. An acceptable ontology enrichment algorithm at least has to fulfill the criteria of a high inner-ontological and a low outer-ontological enrichment tendency. In this paper we introduced such measures - which are based on a random reduction of the concept hierarchy in a given ontology. We briefly sketched an algorithm, which was configured by applying the overlap measures. For other approaches applied to the task our proposal of overlap and outer-ontological would work, as long as the approach produces an outcome in the sense of our definition of ontology enrichment. In contrast the formalization of the relevance measures we presented is more restricted to the needs of similarity-based ontology enrichment approaches, which come along with the need of an additional candidate detection mechanism. Altogether overlap and misclassification measures only consider the ontology, the ontology enrichment algorithm and the text corpora for an evaluation, whereas the relevance measures presented here need extra knowledge on lexical variants of concept descriptors.
In our future work we will examine the correlation between the different variants of the overlap and the relevance measures we presented. Another ambitious but helpful task would be a study of correlations to user-centric ontology enrichment evaluations.

References

[AL01] E. Aguirre, M. Lersundi: Extraccion de relaciones lexico-semanticas a partir de palabras derivadas usando patrones de definicion. Procesamiento del Language Natural **27**, 2001

[BR99] R. Baeza-Yates, B. Ribeiro-Neto: Modern Information Retrieval, Addison Wesley, 1999.

[BNC00] G. Bisson, C. Nedellec, L. Canamero: Designing clustering methods for ontology building - The Mo'K workbench, Proceedings of the ECAI-2000 workshop on Ontology Learning, Berlin, Germany, 2000

[BEH+02] E. Bozsak, M. Ehrig, S. Handschuh et al.: KAON: Towards a Large-Scale Semantic Web, In: Bauknecht, K.; Min Tjoa, A.; Quirchmayr, G. (Eds.): Proc. of the 3rd Intl. Conf. on E-Commerce and Web Technologies (EC-Web 2002), 2002

[CVB+02] O. Chapelle, V. Vapnik, O. Bousquet, S. Mukherjee: Choosing Multiple Parameters for Support Vector Machines. Machine Learning **46**(1-3): 131-159 (2002)

[EN94] R. ElMasri, S. B. Navathe: Fundamentals of Database Systems, 2nd Edition. Benjamin/Cummings 1994

[F04] A. Faatz: Ein Verfahren zur Anreicherung fachgebietsspezifischer Ontologien durch Begriffsvorschläge, PhD thesis, Darmstadt University of Technology, 2004

[FS02] A. Faatz, R. Steinmetz: Ontology Enrichment with Texts from the WWW, Proceedings of the First International Workshop on Semantic Web Mining, European Conference on Machine Learning 2002, Helsinki 2002

[FHS+02] A. Faatz, S. Hoermann, C. Seeberg, R. Steinmetz: Conceptual Enrichment of Ontologies by means of a generic and configurable approach. In Proceedings of the ESSLLI 2001 Workshop on Semantic Knowledge Acquisition and Categorisation, August 2001

[GGW01] P. Ganesan, H. Garcia-Molina, J. Widom: Exploiting Hierarchical Domain Structure to Compute Similarity, Extended Technical Report, 2001

[GFC04] A. Gomez-Perez, M. Fernandez-Lopez, O. Corcho: Ontological Engineering, Springer Verlag, 2004

[G92] G. Grefenstette: Use of Syntactic Context to Produce Term Association Lists for Text Retrieval, SIGIR 1992, Copenhagen, Denmark, 1992

[Gr93] T. Gruber: A translation approach to portable ontology specifications. Knowledge Acquisition, 5(2):199–220, 1993

[GW02] N. Guarino, C. A. Welty: Evaluating ontological decisions with OntoClean, Communications of the ACM **45**(2), 2002

[HJ02] C. W. Holsapple, K. D. Joshi: A collaborative approach to ontology design. Commun. ACM **45**(2): 42-47 ,2002

[H98] I. Horrocks: Using an expressive description logic: FaCT or fiction? Principles of Knowledge Representation and Reasoning: Proceedings of the Sixth International Conference (KR'98), San Francisco, California, USA, 1998

[L99] L. Lee: Measures of Distributional Similarity , 37th Annual Meeting of the Association for Computational Linguistics, College Park, Maryland, USA, 1999

[Li03] Y. Li, Z.A. Bandar, D. McLean: An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources, IEEE Transactions on Knowlege and Data Engineering **15**, 2003

[MS02] A. Maedche, S. Staab: Measuring Similarity between Ontologies, Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web, 13th International Conference, EKAW 2002, Siguenza, Spain, 2002

[MS01] A. Maedche, S. Staab: Ontology Learning for the Semantic Web IEEE Intelligent Systems **16** (2), Special Issue on Semantic Web, 2001

[MPS02] A. Maedche, V. Pekar, S. Staab: Ontology Learning Part One - On Discoverying Taxonomic Relations from the Web, in Ning Zhong et al. (eds) Web Intelligence. Springer, 2002

[NVC+04] R. Navigli, P. Velardi, A. Cucchiarelli, F. Neri: Automatic Ontology Learning: Supporting a Per-Concept Evaluation by Domain Experts, In Proceedings of the ECAI-2004 Workshop on Ontology Learning and Population (ECAI-OLP), Valencia, Spain, August 2004.

[N99] C. Nedellec: Corpus-Based Learning of Semantic Relations by the ILP System Asium, Learning Language in Logic, Springer, 1999

[Q98] U. Quasthoff: Tools for Automatic Lexicon Maintenance: Acquisition, Error Correction, and the Generation of Missing Values. In: Proceedings of the first International Conference on Language Resources and Evaluation, LREC 1998

[R99] P. Resnik: Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language, Journal of Artificial Intelligence Research **11**, 1999

[S04] M. Sabou: Extracting Ontologies from Software Documentation: a Semi-Automatic Method and its Evaluation, In Proceedings of the ECAI-2004 Workshop on Ontology Learning and Population (ECAI-OLP), Valencia, Spain, August 2004.

[SG83] G. Salton, M.J. McGill: Introduction To Modern Information Retrieval, McGraw Hill, New York, 1983

[SM01] G. Stumme, A. Maedche: FCA-MERGE: Bottom-Up Merging of Ontologies, Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, IJCAI 2001, Seattle, Washington, USA, August 4-10, 2001

[TRB02] A. Todirascu, L. Romary, D. Bekhouche: Vulcain - An Ontology-Based Information Extraction System, Proceedings of: Natural Language Processing and Information Systems, 6th International Conference on Applications of Natural Language to Information Systems, NLDB 2002, Stockholm, Sweden, June 27-28, 2002

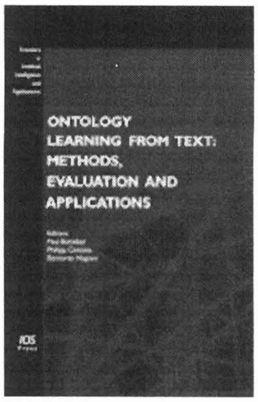
[WordNet] WordNet, a lexical database for the English language, *http://www.cogsci.princeton.edu/ wn/*

[wort] The German Wortschatz Project, *www.wortschatz.de*

[Y92] D. Yarowsky: Word-Sense Disambiguation using Statistical Models of Roget's Categories Trained on Large Corpora, Proceedings of COLING-92, Nantes, France,1992

>

Ontology Learning from Text: Methods, Evaluation and Applications



Volume 123 Frontiers in Artificial Intelligence and Applications
Edited by: P. Buitelaar, P. Cimiano and B. Magnini
July 2005, 180 pp., hardcover
ISBN: 1-58603-523-1
Price: US$102 / €85 / £59

Order Online

Download for this book

This volume brings together ontology learning, knowledge acquisition and other related topics. It presents current research in ontology learning, addressing three perspectives. The first perspective looks at *methodologies* that have been proposed to automatically extract information from texts and to give a structured organization to such knowledge, including approaches based on machine learning techniques. Then there are *evaluation* methods for ontology learning, aiming at defining procedures and metrics for a quantitative evaluation of the ontology learning task; and finally *application* scenarios that make ontology learning a challenging area in the context of real applications such as bio-informatics. According to the three perspectives mentioned above, the book is divided into three sections, each including a selection of papers addressing respectively the methods, the applications and the evaluation of ontology learning approaches.

***IOS* Press**
Nieuwe Hemweg 6B, 1013 BG Amsterdam, The Netherlands
Tel.: +31 20 688 3355, Fax: +31 20 687 0039
E-mail: info@iospress.nl

420~~B~~ : 15 = 28
30
―――
120