

On Automatic Modeling and Use of Domain-specific Ontologies

Troels Andreassen, Henrik Bulskov, and Rasmus Knappe

Department of Computer Science,
Roskilde University,
P.O. Box 260, DK-4000 Roskilde, Denmark
{troels,bulskov,knappe}@ruc.dk

Abstract. In this paper, we firstly introduce an approach to the modeling of a domain-specific ontology for use in connection with a given document collection. Secondly, we present a methodology for deriving conceptual similarity from the domain-specific ontology. Adopted for ontology representation is a specific lattice-based concept algebraic language by which ontologies are inherently generative. The modeling of a domain specific ontology is based on a general ontology built upon common knowledge resources as dictionaries and thesauri. Based on analysis of concept occurrences in the object document collection the general ontology is restricted to a domain specific ontology encompassing concepts instantiated in the collection. The resulting domain specific ontology and similarity can be applied for surveying the collection through key concepts and conceptual relations and provides a means for topic-based navigation. Finally, a measure of concept similarity is derived from the domain specific ontology based on occurrences, commonalities, and distances in the ontology.

1 Introduction

The use of ontologies can contribute significantly to the structure and organization of concepts and relations within a knowledge domain.

The introduction of ontologies into tools for information access provides foundation for enhanced, knowledge-based approaches to surveying, indexing and querying of document collections.

We introduce in this paper the notion of an *instantiated ontology*, as a subontology derived from a general ontology and restricted by the set of instantiated concepts in a target document collection. As such, this instantiated ontology represents a conceptual organization reflecting the document collection, and reveals domain knowledge, for instance about the thematic areas of the domain which in turn facilitates means for a topic-based navigation and visualization of the structure within the given domain.

The primary focus of this paper concerns the modeling and use of ontologies. We introduce, in section 2, to a formalism for representation of ontologies. Section 3 describes the modeling of general and instantiated ontologies, respectively

and illustrates the use of instantiated ontologies for surveying, visualization of both a given domain, but also queries or sets of domain concepts. Finally, in section 4, we derive a measure of conceptual similarity, based on the structure and relations of the ontology.

Both modeling and use of ontologies relies on the possibility to identify concept occurrences in – or generate conceptual descriptions of – text. To this end we assume a processing of text by a simplified natural language parser providing concept occurrences in the text. This parser may be applied for indexing of documents as well as for interpreting queries. The most simplified principle used here is basically a nominal phrase bracketing and an extract of nouns and adjectives that are combined by a “*noun CHR adjective*”-pattern concept (CHR representing a “characterized by” relation).

Thus, for instance, for the sentence *Borges has been widely hailed as the foremost contemporary Spanish-American writer.* the parser may produce the following:

borges, writer[CHR:*contemporary*, CHR:*foremost*, CHR:*spanish_american*]

Concept expressions, that are the key to modelling and use of ontologies, are explained in more detail below, we refer, however, to [2] for a discussion of general principles behind parsing for concepts.

2 Representation of Ontologies

The purpose of the ontology is to define and relate concepts that may appear in the document collection or in queries to this.

We define a generative ontology framework where a basis ontology situates a set of atomic term concepts \mathbf{A} in a concept inclusion lattice. A concept language (description language) defines a set of well-formed concepts, including both atomic and compound term concepts.

The concept language used here, ONTOLOG[8], defines a set of semantic relations \mathbf{R} that can be used for “attribution” (feature-attachment) of concepts to form compound concepts. The set of available relations may vary with different domains and applications. We may choose $\mathbf{R} = \{\text{WRT, CHR, CBY, TMP, LOC, \dots}\}$, for *with respect to, characterized by, caused by, temporal, location*, respectively.

Expressions in ONTOLOG are concepts situated in the ontology formed by an algebraic lattice with concept inclusion (ISA) as the ordering relation.

Attribution of concepts – combining atomic concepts into compound concepts by attaching attributes – can be written as feature structures. Simple attribution of a concept c_1 with relation r and a concept c_2 is denoted $c_1[r: c_2]$.

Given atomic concepts \mathbf{A} and relations \mathbf{R} , the set of well-formed terms \mathbf{L} of the ONTOLOG language is defined as follows.

- if $x \in \mathbf{A}$ then $x \in \mathbf{L}$
- if $x \in \mathbf{L}$, $r_i \in \mathbf{R}$ and $y_i \in \mathbf{L}$, $i = 1, \dots, n$
then $x[r_1: y_1, \dots, r_n: y_n] \in \mathbf{L}$

It appears that compound terms can be built from nesting, for instance, $c_1[r_1 : c_2[r_2 : c_3]]$ and from multiple attribution as in $c_1[r_1 : c_2, r_2 : c_3]$.

The attributes of a term with multiple attributes $T = x[r_1 : y_1, \dots, r_n : y_n]$ are considered as a set, thus we can rewrite T with any permutation of $\{r_1 : y_1, \dots, r_n : y_n\}$.

3 Modeling Ontologies

One objective in the modelling of domain knowledge is for the domain expert or knowledge engineer to identify significant concepts in the domain.

Ontology modelling in the present context is, compared to other works within the ontology area, a limited approach. The modelling consists of two parts. Firstly an inclusion of knowledge from available knowledge sources into a general ontology and secondly a restriction to a domain-specific part of the general ontology. The first part involves modeling of concepts in a generative ontology using different knowledge sources. In the second part a domain-specific ontology is retrieved as a subontology of the general ontology. The restriction to this subontology is build based on the set of concepts that appears (is instantiated) in the document collection and the result is called an instantiated ontology.

3.1 The General Ontology

Sources for knowledge base ontologies may have various forms. Typically a taxonomy can be supplemented with for instance word and term lists as well as dictionaries for definition of vocabularies and for handling of morphology.

We will not go into details on the modeling here but just assume the presence of a taxonomy in the form of a simple taxonomic concept inclusion relation ISA_{KB} over the set of atomic concepts \mathbf{A} . ISA_{KB} and \mathbf{A} expresses the domain and world knowledge provided. ISA_{KB} is assumed to be explicitly specified – e.g. by domain experts – and would most typically not be transitively closed.

Based on $\widehat{\text{ISA}}_{\text{KB}}$, the transitive closure of ISA_{KB} , we can generalize into a relation over all well-formed terms of the language \mathbf{L} by the following:

- if $x \widehat{\text{ISA}}_{\text{KB}} y$ then $x \leq y$
- if $x[\dots] \leq y[\dots]$ then also
 - $x[\dots, r : z] \leq y[\dots]$, and
 - $x[\dots, r : z] \leq y[\dots, r : z]$,
- if $x \leq y$ then also
 - $z[\dots, r : x] \leq z[\dots, r : y]$

where repeated \dots in each case denote zero or more attributes of the form $r_i : w_i$.

The general ontology $O = (\mathbf{L}, \leq, \mathbf{R})$ thus encompasses a set of well-formed expressions \mathbf{L} derived from the concept language with a set of atomic concepts \mathbf{A} , an inclusion relation generalized from an expert provided relation ISA_{KB} and a supplementary set of semantic relations \mathbf{R} , where for $r \in \mathbf{R}$ we obviously have that $x[r : y] \leq x$ and that $x[r : y]$ is in relation r to y . Observe that \mathbf{L} is infinite and that O thus is generative.

3.2 The Domain-specific Ontology

Apart from the general ontology O , the target document collection contributes to the construction of the domain ontology. We assume a processing of the target document collection, where an indexing of text in documents, formed by sets of concepts from \mathbf{L} , is attached. In broad terms the domain ontology is a restriction of the general ontology to the concepts appearing in the target document collection.

More specifically the generative ontology is, by means of concept occurrence analysis over the document collection, transformed into a domain specific ontology restricted to include only the concepts instantiated in the documents covering that particular domain. We thus introduce the domain specific ontology as an “instantiated ontology” of the general ontology with respect to the target document collection.

The instantiated ontology $O_{\hat{I}}$ appears from the set of all instantiated concepts I , firstly by expanding I to \hat{I} – the transitive closure of the set of terms and subterms of term in I – and secondly by producing the subontology consisting of \hat{I} connected by relations from O between elements of \hat{I} .

The subterms of a term c is obtained by the decomposition $\tau(c)$. $\tau(c)$ is defined as the set of all subterms of c , which thus includes c and all attributes of subsuming concepts for c .

$$t(c) = \{x|c \leq x[\dots, r: y] \vee c \leq y[\dots, r: x], x, y \in \mathbf{L}, r \in \mathbf{R}\}$$

$$\tau(c) = \text{closure of } \{c\} \text{ with respect to } t$$

For a set of terms we define $\tau(C) = \bigcup_{c \in C} \tau(c)$. As an example, we have that

$$\tau(c_1[r_1: c_2[r_2: c_3]]) = \{c_1[r_1: c_2[r_2: c_3]], c_1[r_1: c_2], c_1, c_2[r_2: c_3], c_2, c_3\}.$$

Let $\omega(C)$ for a set of terms C be the transitive closure of C with respect to \leq . Then the expansion of the set of instantiated concepts I becomes:

$$\hat{I} = \omega(\tau(I))$$

Now, the C -restriction subontology $O_C = (C, \leq, \mathbf{R})$ with respect to a given set of concepts C , is the subontology of O over concepts in C connected by \leq and \mathbf{R} . Thus the instantiated ontology $O_{\hat{I}} = (\hat{I}, \leq, \mathbf{R}) = (\omega(\tau(I)), \leq, \mathbf{R})$ is the \hat{I} -restriction subontology of O .

Finally we define ISA as the transitive reduction of \leq and consider $(\hat{I}, \text{ISA}, \mathbf{R})$ for visualization and as basis for similarity computation below.

3.3 Modeling Domain-specific Ontologies – An Example

Consider the knowledge base ontology ISA_{KB} shown in Figure 1a. In this case we have $\mathbf{A} = \{cat, dog, bird, black, brown, red, animal, color, noise, anything\}$

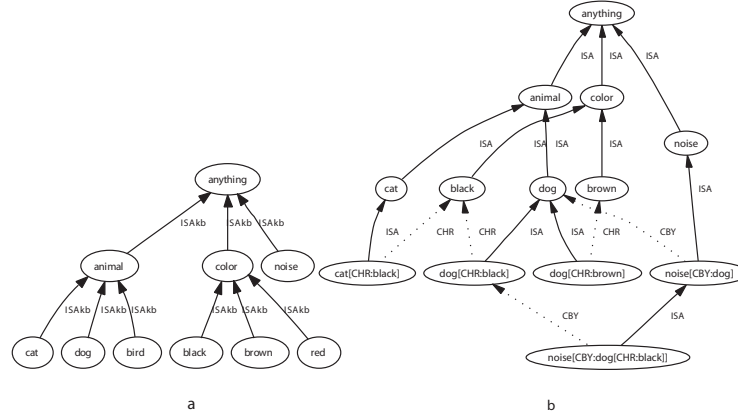


Fig. 1. a) An example knowledge base ontology ISA_{KB} **b)** A simple instantiated ontology based on figure a and the set of instantiated concepts $cat[CHR:black]$, $dog[CHR:black]$, $dog[CHR:brown]$, $noise[CBY:dog[CHR:black]]$.

and \mathbf{L} includes \mathbf{A} and any combination of compound terms combining elements of \mathbf{A} with attributes from \mathbf{A} by relations from \mathbf{R} , due to the generative quality of the ontology. Now assume a miniature target document collection with the following instantiated concepts:

$$I = cat[CHR:black], dog[CHR:black], dog[CHR:brown], noise[CBY:dog[CHR:black]]$$

The decomposition $\tau(I)$ includes any subterm of elements from I , while $\hat{I} = \omega(\tau(I))$ adds the subsuming $\{animal, color, anything\}$:

$$\hat{I} = \{cat, dog, black, brown, animal, color, noise, anything, cat[CHR:black], dog[CHR:black], dog[CHR:brown], noise[CBY:dog], noise[CBY:dog[CHR:black]]\}$$

where the concepts red and $bird$ from \mathbf{A} are omitted because they are not instantiated.

The resulting instantiated ontology $(\hat{I}, \leq, \mathbf{R})$ is transitively reduced into the domain-specific ontology $(\hat{I}, ISA, \mathbf{R})$ as shown in figure 1b.

3.4 Visualization of Instantiated Ontologies

As the instantiated ontology is a restriction of a general ontology with respect to a set of concepts, it can be used for providing structured descriptions. The restriction could be with respect to the sets of concepts in a particular target document collection. But it could also comprise the set of concept of a query,

the set of concept in a complete search result, or any set of concept selected by the user or the querying system.

The notion of instantiated ontologies has as such applications with respect to navigation and surveying of the topics covered by the domain in question, where the domain could be the instantiated ontology of any of the suggested restrictions above.

As the concepts instantiated in the document collection expresses a structuring of the information available, we can it to address one of the difficulties users have to overcome when querying information systems, which concerns the transformation of their information need into the descriptions used by the system.

Consider the following example, where we have a document collection with the following four instantiated concepts

$$I = \{stockade[CHR:old], rampart[CHR:old], church[CHR:old], palisade\}$$

and a global ontology constructed from version 1.6 of the WordNet lexicon [6], the Suggested Upper Merged Ontology (SUMO) [7], and the mid-level ontology (MILO) [7] designed to bridge the high-level ontology SUMO and WordNet.

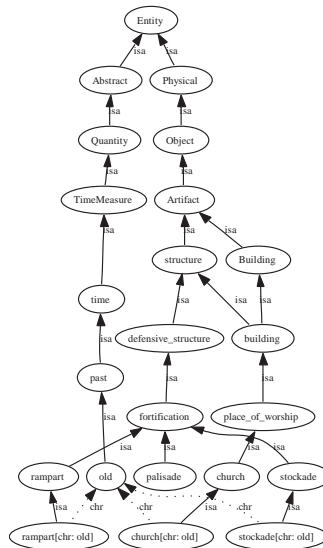


Fig. 2. A simple instantiated ontology, based on WordNet, SUMO and MILO and the four concepts *stockade*[CHR:old], *rampart*[CHR:old], *church*[CHR:old], *palisade*

The instantiated ontology reveals two different aspects covered by the document collection, 1) different kinds of fortifications and 2) a place of worship. On a more general level the instantiated ontology describes buildings and the abstract notion of something dated back in time.

Another use of instantiated ontologies is for visualizing user queries. When users pose queries to the system using polysemous concepts, the instantiated ontology constructed from the query can be used to visualize the different senses known to the system. If for example a user poses a query $Q = \{bank, huge\}$, then the system cannot use the concept *huge* to resolve the context/diambiguate *bank*, since *huge* can be used in connection with different senses of *bank*.

One possible way to incorporate the knowledge visualized is to the user to identify which sense meant, and use the disambiguated concept in the query evaluation.

4 Deriving Similarity

As touched upon elsewhere in this paper a domain ontology, that reflects a document collection, may provide an excellent means to survey and give perspective to the collection. However as far as access to documents is concerned ontology reasoning is not the most obvious evaluation strategy and it may well entail scaling problems. Applying measures of similarity derived from the ontology is a way to replace reasoning with simple computation still influenced by the ontology. A well-known and straightforward approach to this is the shortest path approach [5,9], where closeness between two concepts in the ontology imply high similarity. A problem with this approach is that multiple connections are ignored.

Consider the following example, where we have three concepts, all of which are assumed subclasses of pet: *dog*[CHR:grey], *cat*[CHR:grey] and *bird*[CHR:yellow]. If we consider only the ISA-edges then there is no difference in similarity between any pair of them due to the fact that they are all specializations of pet. If we on the other hand also consider the semantic edges (CHR), then we can add the aspect of shared attribution to the computation of similarity. In our example, we can, by including the the CHR-edges, capture the intuitive difference in similarity between two grey pets compared to the similarity between a grey and a yellow pet. This difference would be visualized by the existence of a path, that includes the shared concept, between the two concepts sharing attribution.

4.1 Shared Nodes Similarity

To differentiate here an option is to consider all paths rather than only the shortest path. A “shared nodes” approach that reflects multiple paths, but still avoids the obvious complexity of full computation of all paths is presented in [1]. In this approach the basis for the similarity between two concepts c_1 and c_2 is the set of “upwards reachable” concepts (nodes) shared between c_1 and c_2 . This is, with $\alpha(x) = \omega(\tau(x))$, the intersection $\alpha(x) \cap \alpha(y)$.

Similarity can be defined in various ways, one option being, as described in [3], a weighted average, where $\rho \in [0, 1]$ determines the degree of influence of the nodes reachable from x respectively y .

$$sim(x, y) = \rho \frac{|\alpha(x) \cap \alpha(y)|}{|\alpha(x)|} + (1 - \rho) \frac{|\alpha(x) \cap \alpha(y)|}{|\alpha(y)|} \quad (1)$$

As it appears the upwards expansion $\alpha(c)$ includes not only all subsuming concepts $\{c_i \mid c \leq c_i\}$ but also concepts that appears as direct or nested attributes to c or to any subsuming concept of these attributes. The latter must be included if we want to cope with multiple connections and want to consider for instance two concepts more similar if they bear the same color.

4.2 Weighted Shared Nodes Similarity

However, a further refinement seems appropriate here. If we want two concepts to be more similar if they have an immediate subsuming concept (e.g. $cat[CHR:black]$ and $cat[CHR:brown]$ due to the subsuming cat) than if they only share an attribute (e.g. $black$ shared by $cat[CHR:black]$ and $dog[CHR:black]$) we must differentiate and cannot just define $\alpha(c)$ as a crisp set. The following is a generalization to fuzzy set based similarity.

First of all notice that $\alpha(c)$ can be derived as follows. Let the triple (x, y, r) be the edge of type r from concept x to concept y , E be the set of all edges in the ontology, and T be the top concept. Then we have:

$$\begin{aligned}\alpha(T) &= \{T\} \\ \alpha(c) &= \{c\} \cup (\cup_{(c,c_i,r) \in E} \alpha(c_i))\end{aligned}$$

A simple modification that generalizes $\alpha(c)$ to a fuzzy set is obtained through a function $weight(r)$ that attach a weight to each relation type r . With this function we can generalize to:

$$\begin{aligned}\alpha(T) &= \{1/T\} \\ \alpha(c) &= \{c\} \cup (\cup_{(c,c_i,r) \in E} weight(r) * \alpha(c_i)) \\ &= \{c\} \cup (\cup_{(c,c_i,r) \in E} \sum_{\mu(c_{ij})/c_{ij} \in \alpha(c_i)} weight(r) * \mu(c_{ij})/c_{ij})\end{aligned}$$

$\alpha(c)$ is thus the fuzzy set of nodes reachable from the concept c and modified by weights of relations $weight(r)$. For instance from the instantiated ontology in figure 1b, assuming relation weights $weight(ISA) = 1$, $weight(CHR) = 0.5$ and $weight(CBY) = 0.5$, we have:

$$\begin{aligned}\alpha(noise[CBY:dog[CHR:black]]) &= 1/noise[CBY:dog[CHR:black]] + 1/noise + 0.5/dog[CHR:black] + \\ &0.5/dog + 0.5/animal + 0.25/black + 0.25/color + 1/anything\end{aligned}$$

For concept similarity we can still use the parameterized expression (1) above, applying minimum for fuzzy intersection and sum for fuzzy cardinality:

$$\alpha(cat[CHR:black]) \cap \alpha(dog[CHR:black]) = 0.5/black + 0.5/color + 1/animal + 1/anything$$

$$|\alpha(cat[CHR:black]) \cap \alpha(dog[CHR:black])| = 3.0$$

The similarities between $dog[CHR:black]$ and other concepts in the ontology are, when collected in a fuzzy subset of similar concepts (with $similar(x) = \sum sim(x, y)/y$) and $\rho = \frac{4}{5}$ the following:

$$\begin{aligned}similar(dog[CHR:black]) &= 1.00/dog[CHR:black] + 0.68/dog + 0.6/cat[CHR:black] + \\ &0.6/noise[CBY:dog[CHR:black]] + 0.52/animal + 0.45/black + 0.45/cat + 0.39/color + \\ &0.36/anything + 0.34/brown + 0.26/noise\end{aligned}$$

5 Conclusion

Firstly, we have introduced the notion of a domain-specific ontology as a restriction of a general ontology to the concepts instantiated in a document collection, and we have demonstrated its applications with respect to navigation and surveying of a target document collection.

Secondly, we have presented a methodology for deriving similarity using the domain-specific ontology by means of weighted shared nodes. The proposed measure incorporates multiple aspects when calculating overall similarity between concepts, but also respects the structure and relations of the ontology.

It should be noted that modelling of similarity functions is far from objective. It is not possible to define optimal functions, neither in the general nor in the domain specific case. The best we can do is to specify flexible, parameterized functions on the basis of obvious 'intrinsic' properties of intuitive interpretation, and then to adjust and evaluate these functions on an empirical basis.

References

1. Andreasen, T., Bulskov, H. and Knappe, R.: On Querying Ontologies and Databases, Flexible Query Answering Systems, 6th International Conference, FQAS 2004, Lyon, France, June 24-26, 2004, Proceedings
2. Andreasen, T.; Jensen, P. Anker; Nilsson, J. Fischer; Paggio, P.; Pedersen, B.S.; Thomsen, H. Erdman: Content-based Text Querying with Ontological Descriptors, in Data & Knowledge Engineering 48 (2004) pp 199-219, Elsevier, 2004.
3. Andreasen, T., Bulskov, H., and Knappe, R.: Similarity from Conceptual Relations, pp. 179–184 in Ellen Walker (Eds.): 22nd International Conference of the North American Fuzzy Information Processing Society, NAFIPS 2003, Chicago, Illinois USA, July 24–26, 2003, Proceedings
4. Beckwith, R.; Miller, G. A. & Teng, R. (Eds.): Design and implementation of the WordNet lexical database and searching software, <http://www.cogsci.princeton.edu/wn/5papers.ps>.
5. Bulskov, H.; Knappe, R. & Andreasen, T.: On Measuring Similarity for Conceptual Querying, LNAI 2522, pp. 100–111 in Andreasen T.; Motro A.; Christiansen H.; Larsen H.L.(Eds.): Flexible Query Answering Systems 5th International Conference, FQAS 2002, Copenhagen, Denmark, October 27–29, 2002, Proceedings
6. Miller, George: WordNet: An On-line Lexical Database, International Journal of Lexicography, Volume 3, Number 4, 1990
7. Niles, I.; Pease, A.: Towards a Standard Upper Ontology, in Chris Welty and Barry Smith (eds.): Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001), Ogunquit, Maine, October 17-19, 2001.
8. Nilsson, J. Fischer: A Logico-algebraic Framework for Ontologies – ONTOLOG, in Jensen, P. Anker & Skadhauge, P. (eds.): Proceedings of the First International OntoQuery Workshop – Ontology-based interpretation of NP's, Department of Business Communication and Information Science, University of Southern Denmark, Kolding, 2001
9. Rada, Roy; Mili, Hamed; Bicknell, Ellen & Blettner, Maria: Development and Application of a Metric on Semantic Nets, IEEE Transactions on Systems, Man, and Cybernetics, Volume 19, Number 1, pp. 17–30, 1989