

Searching Ontologies Based on Content: Experiments in the Biomedical Domain

Harith Alani

Electronics and Computer
Science
University of Southampton
Southampton, UK
h.alani@ecs.soton.ac.uk

Natalya F. Noy

Stanford Medical Informatics
Stanford University
Stanford, CA, USA
noy@stanford.edu

Nigam Shah

Stanford Medical Informatics
Stanford University
Stanford, CA, USA
nigam@stanford.edu

Nigel Shadbolt

Electronics and Computer
Science
University of Southampton
Southampton, UK
nrs@ecs.soton.ac.uk

Mark A. Musen

Stanford Medical Informatics
Stanford University
Stanford, CA, USA
musen@stanford.edu

ABSTRACT

As more ontologies become publicly available, finding the “right” ontologies becomes much harder. In this paper, we address the problem of **ontology search**: finding a collection of ontologies from an ontology repository that are relevant to the user’s query. In particular, we look at the case when users search for ontologies relevant to a particular *topic* (e.g., an ontology about anatomy). Ontologies that are most relevant to such query often do not have the query term in the names of their concepts (e.g., the Foundational Model of Anatomy ontology does not have the term “anatomy” in any of its concepts’ names). Thus, we present a new ontology-search technique that helps users in these types of searches. When looking for ontologies on a particular topic (e.g., *anatomy*), we retrieve from the Web a collection of terms that represent the given domain (e.g., terms such as *body*, *brain*, *skin*, etc. for *anatomy*). We then use these terms to expand the user query. We evaluate our algorithm on queries for topics in the biomedical domain against a repository of biomedical ontologies. We use the results obtained from experts in the biomedical-ontology domain as the gold standard. Our experiments demonstrate that using our method for query expansion improves retrieval results by a 113%, compared to the tools that search only for

the user query terms and consider only class and property names (like Swoogle). We show 43% improvement for the case where not only class and property names but also property values are taken into account.

Categories and Subject Descriptors

H.3.3 [Information Search and Retrieval]: Selection; I.2.4 [Knowledge Representation Formalisms and Methods]: Representation languages, Semantic networks

General Terms

Algorithms, Experimentation, Human Factors

Keywords

Ontology searching, Biomedical ontologies, Ontology analysis

1. DEFINING ONTOLOGY SEARCH

Ontologies are the key component of the Semantic Web. Today, an ever growing number of ontologies in various domains is becoming available. At the time of this writing, Swoogle¹ boasts over 10,000 ontologies of various types, sizes, and qualities. However, the more ontologies are available, the harder it is for users to find ontologies relevant to their domain of interest. One would expect that typing “anatomy” in the search field of an ontology-search engine should get the user the Foundational Model of Anatomy (FMA) [12]—one of the most popular anatomy ontologies—as one of the top results. Yet, the FMA does not have a single class with the word “anatomy” in its name. Therefore, a keyword search on Swoogle for “anatomy” will not actually return the

FMA. This situation is quite common: ontologies relevant to a particular topic or domain often do not contain the name of the topic or domain itself in names of their classes and properties and often do not contain it in property values as well.

We define the problem of **ontology search** in an ontology repository R as follows: given a user query, return a collection of ontologies from R that are relevant to the topic of the query. We understand that for our ontology-search methods to be truly useful for users, we must combine them with promising methods for ontology evaluation and ranking. We view the two problems—ontology search and ontology ranking—as two complementary sides of the problem of finding relevant ontologies. We address the problem of ontology search in this paper.

To understand better how users tend to search for ontologies, we monitored the user mailing lists of Protégé,² a widely used ontology-editing tool. Protégé mailing lists often receive requests from users seeking ontologies for particular domains. We observed that almost all such user requests name the domain (e.g. History, Economy, Algebra), but not the representative terms for the domain. Search engines usually return only the ontologies that have the query term itself in their class or property names, rather than searching for the ontologies that cover the *domain* described by the query term.

One way to find out if an ontology covers a particular domain is through *ontology metadata* provided by ontology authors. Ontology repositories, such as the BioPortal by the National Center for Biomedical Ontology³ enable authors to specify the domain of their ontology and other metadata. However, most ontologies do not contain this type of metadata and there is no standard or widely accepted way of specifying it.

In this paper, we present a new mechanism for ontology search that adds the element of domain knowledge to the process. We use the Web itself to expand the user query with terms that are representative of the topic. We collect these terms from the Web pages returned by a Web search with the user query. In a sense, we are mimicking the way a human expert would go about finding terms relevant to a particular topic: search the Web pages (or an encyclopedia, such as Wikipedia) for that topic and identify terms that are relevant for that topic. Thus, when looking for pages on “anatomy”, we find that the following terms are representative of this topic: *body*, *brain*, *skin*, *bone*, *eye*, *neck*, and so on. Then we use these terms to query the ontology repository.

In order to evaluate the results of our approach, we

²<http://protege.stanford.edu>

³<http://www.bioontology.org/ncbo/faces/index.xhtml>

asked several experts in biomedical ontologies to identify ontologies from a repository that are relevant to several specific queries. We then used the results produced by domain experts as the gold standard and compared compared the retrieval results produced by our approach with the results produced by two baseline methods: (1) search the class and property names of ontologies in the repository using only the query terms specified by the user (this type of search is what most ontology-search engines do) and (2) search not only class and property names but also property values (e.g., synonyms, definitions, etc.) the query terms specified by the user (this type of search is available in BioPortal). Our experiments demonstrated a 113% improvement (in terms of F-measure) over the former baseline methods (search class and property labels) and 43% improvement over the latter (add property values to the search).

More specifically, this paper makes the following contributions:

- We analyze the inter-expert agreement for the problem of ontology search.
- We show that searching in property values and not only class and property labels produces significant improvement in search results.
- We describe an approach to ontology search that uses Wikipedia for query expansion.
- We evaluate the approach on a repository of biomedical ontologies, by comparing the results to those generated manually by domain experts.

Our evaluation shows that the approach is very promising for queries where users are searching for ontologies on a specific topic. As such, we envision such approach as an essential component of a broader system for ontology search and evaluation.

2. RELATED WORK

The problem of finding relevant documents based on a set of query terms that the user provides is not a new one. This problem is at the core of the Web search in general. It is fair to say that PageRank [9] revolutionised Web search by introducing a new algorithm that was well tuned for the structure of the Web. However, ontologies differ from standard Web pages in structure as well as purpose. Web pages are usually made up of text, while ontologies are highly structured graphs of classes and properties. Web pages describe and communicate information to humans, while ontologies represent domain models. Nevertheless, existing ontology-search engines tend to apply traditional Web search techniques when searching for ontologies.

Swoogle [6] is currently the dominant engine for searching ontologies. It searches a large index of ontologies crawled off the Web for classes and properties with labels containing the keywords entered by the user.

OntoSearch [10] employs reasoning using Pellet ⁴ in ontology search. Users submit SPARQL queries to a metaontology to search the contents of the ontologies that OntoSearch stores. OntoSearch supports only direct matches with labels or structure.

OntoSelect [4] is an ontology search engine that provides a content-based search, similar to our earlier work described in [3]. Users of OntoSelect submit a URL for a single Web page to be treated as a corpus, from which OntoSelect extracts the top 20 most frequent terms to expand the query. What distinguishes OntoSelect from the approach we are proposing in this paper is that while they rely on the user to find a one-document corpus, we are moving towards automating this process to mimic the way users tend to search for ontologies.

Some work on ontology search limited the search to specific *triples*, rather than searching for ontologies as a whole [13]. Any ontology that contains the triple in question is returned. Such approach can be useful when dealing with applications with very specific and limited search requirements, such as to find only a given triple (e.g. *(project, related to, researcher)*), but it is not suitable to perform a general search for domain ontologies, where the need is for a whole ontology, rather than a single triple.

When traditional retrieval methods do not return sufficient results, *query expansion* is a common approach in Information Retrieval and can be performed following various techniques. For example a term can be expanded based on its taxonomic hierarchy (e.g. broader, narrower terms [11]), or using lexical-semantic relations (e.g. [15]). Another approach is based on statistical analyses (e.g. Vector Space model [14]), where relevant terms are extracted from text based on term frequency analysis (TF). In the work described in this paper, we followed a query-expansion technique based on text analysis.

Ontology ranking is an important complementary problem to the problem of ontology search: after a search engine finds the relevant ontologies, it needs to rank them to indicate which ontologies are more relevant than others. In the future, we envision combining our search techniques with one of the ontology-ranking approaches: the PageRank-based ranking of Swoogle [6], structure-based ranking such as AKTiveRank [1, 2], user ratings [8], and others.

⁴<http://www.mindswap.org/2003/pellet/>

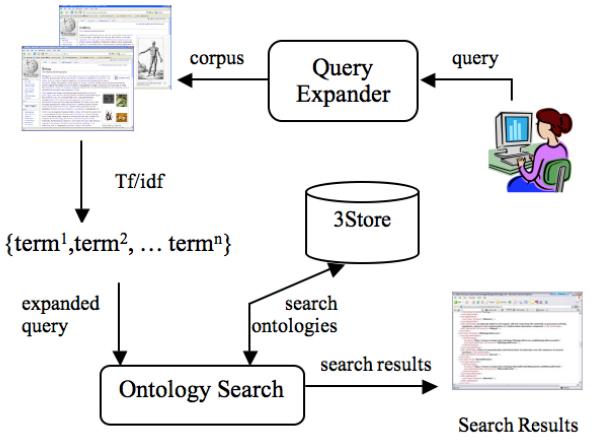


Figure 1: Process of ontology search with query expansion

3. SEARCHING ONTOLOGIES WITH QUERY EXPANSION

In our approach, we treat the query term as a name of a *domain* and use the relevant Web pages to find representative terms with which to expand the query. We can summarize the process as follows: (1) take one or more query terms from the user, (2) expand the query to a set of terms that represent the domain, (3) find ontologies containing the largest number of the terms from the expanded set, taking term frequency into account.

3.1 Expanding the Query

Figure 1 shows the process of query expansion. Once the user submits a query, we locate a number of relevant Web pages to use as a corpus that describes the query domain. To locate the relevant Web pages, we perform a Google search using the query entered by the user. We have experimented with two options for finding the corpus: (1) using the general Google results; and (2) restricting results returned by Google to the Wikipedia site. The second option seemed to produce a better corpus in the biomedical domain (our domain for experiments). Table 1 shows the list of terms extracted for a query on “anatomy” restricting Google search to the Wikipedia site versus a corpus retrieved from Google without any restrictions. Wikipedia pages are useful for query expansion as they provide a good coverage of topic terms [5]. We use Wikipedia results in the rest of this paper.

Another parameter we varied in creating the corpus was the number of pages we take from the results, from 2 to 50 pages. We present results for different values of this parameter (Section 5).

After retrieving the corpus, we calculate the frequency of all the terms that appear in the corpus (excluding stop words) using a simple TF algorithm [14], and select

Wikipedia		Google	
anatomy	anterior	medical	american
posterior	nerve	body	atlas
superior	bone	nemours	available
inferior	grey	institute	anatomy
human	lateral	digestive	muscle
body	season	view	brain
ligament	head	kidney	eye
iris	nail	reproductive	research
bird	neck	medlineplus	heart
eye	hand	health	topic
muscle	hip	female	urinary
colon	organ	import	url
nucleus	fin	nlm	nih
horse	medial	images	gov
gland	foramen	teeth	css
plant	series	connections	tribal
process	episode	diseases	corners
joint	content	education	male
external	color	endocrine	blood
history	change	immune	lung
tissue	structure	texas	medicine
facial	canal	library	diabetes
cell	film	human	lymphatic
encyclopedia	navigation	nervous	bones
registered	artery	joints	nerves

Table 1: Expanding query terms with Wikipedia vs Google using a 2-document corpus. Terms are ordered according to their frequency in the corpus.

the top 50 terms to be used as the new user query. In other words, we expand the user query with additional terms that represent the domain named in the original query. We search the ontologies for all the terms in the expanded query.

To determine if an ontology O in our repository is relevant to the expanded set of query terms T , we determine how many times each query term $t \in T$ appears in the labels of classes, labels of properties, and in property values for datatype properties (e.g., string-valued properties). We normalize the number of occurrences by the TF frequency of the term t in our corpus. For each ontology O , we also remove the term that appears most often in O (the outlier term) from determining the score for O . The latter step of removing the outliers allows us to account for cases where a common term appears hundreds of times in the ontology, but no other terms from the query do. For instance, the term *cell* was part of the query expansion for the query *physiological process*. Many ontologies that were not actually relevant for the query, had hundreds of labels with “cell” in them and hardly any occurrence of any other term from the expanded query. Our assumption here is that an ontology that covers many of the terms, regardless of frequency,

is more relevant than the one that covers one or only a few terms in high frequencies.

More formally, we say that the relevancy score for an ontology O given a set of query terms T , $Score(O, T)$ is the following:

$$Score(O, T) = ClassScore(O, T) + PropertyScore(O, T) + ValueScore(O, T)$$

The $ClassScore(O, T)$ accounts for the class labels in O , property labels contribute to $PropertyScore(O, T)$ and property values contribute to $ValueScore(O, T)$. Specifically:

$$ClassScore(O, T) = \sum_{t \in T} f_t * h(t, O) - \max_{t \in T} (f_t * h(t, O))$$

where f_t is the normalized frequency of the term t in the corpus and $h(t, O)$, the class term hits, is the number of times the term t appears in class labels for classes in ontology O . $\max_{t \in T} (f_t * h(t, O))$ is the score for the term with the highest frequency of occurrence in class labels in O , i.e., the outlier term.

We get $PropertyScore(O, T)$ and $ValueScore(O, T)$ the same way, using property labels and content of property values, respectively.

Once we compute the relevancy scores for all ontologies in a repository, we return the ontologies with the scores above the median value of all non-zero scores. Recall that we address the problem of ontology search and not ontology ranking, and thus we consider all returned ontologies equally relevant for the query. We envision that various ontology-ranking methods (see Section 2) can be applied at this stage to rank the results.

4. EXPERIMENT SETUP

We chose the domain of biomedical ontologies for our empirical evaluation of the algorithm. Focusing search on ontologies covering very similar domains is more challenging than searching through ontologies that represent very different domains. It is usually much easier to filter out an ontology about, say, transport when searching for “anatomy,” than to filter between overlapping biomedical ontologies. For our **ontology repository** R we chose the Open Biomedical Ontologies available through the BioPortal of the National Center for Biomedical Ontologies.⁵ At the time that we performed the experiments, the repository consisted of 55 ontologies representing various biomedical areas. We downloaded the OWL versions of these ontologies and stored them in 3Store [7], which can be accessed with SPARQL queries.

⁵<http://www.bioontology.org/ncbo/faces/index.xhtml>

We used a relatively small ontology repository rather than, say, Swoogle, for several reasons: First, having a limited number of ontologies allowed us to collect high quality input from experts on which ontologies are relevant to the query. Second, knowing what all the ontologies in the repository are and which are relevant, according to the experts, enabled us to get exact figures for recall and precision.

We used the following four **queries** in our experiments: (1) *anatomy*; (2) *pathology*; (3) *physiological process*; (4) *histology*.

4.1 Collecting Data from Experts

We asked 5 experts in the domain of biomedical ontologies to identify the ontologies from our repository that they considered relevant to the four queries above. The specific question that we asked was: “Which of the ontologies listed in the BioPortal represent the knowledge about the following topic?” We gave each expert a spreadsheet with the names of ontologies and four queries and asked them to mark which ontologies are relevant for each query. All the experts were familiar with the ontologies in the repository and also had access to them through the BioPortal. The experts answered our request separately, without conferring with one another.

We then used our algorithm to find ontologies from the repository that were relevant for each query. We compared the results returned by the algorithm with those returned by our experts. If at least one of the experts considered an ontology that our algorithm returned to be relevant for the query, we considered this ontology a hit. The returned ontology was a miss otherwise.

4.2 Baseline Cases

We also created 3 baseline cases with which we compared our results. These cases correspond to basic searches, with no query expansion:

query terms in labels (L): search only class and property labels for the terms in the (non-expanded) query;

query terms in labels and property values (LV): search not only labels but also property values, such as synonyms and comments, for the terms in the non-expanded query;

all ontologies (NULL): return all ontologies in the repository.

The null case obviously provides 100% recall but variable precision, depending on the query.

Note that results in the *L* case are identical to what ontology-search engines such as Swoogle would have returned if their repository was limited to the ontologies

query	number of ontologies
anatomy	21
physiological process	15
pathology	6
histology	21
total	63

Table 2: The number of ontologies identified by experts as relevant to each query (out of 55 ontologies in the repository)

number of experts agreeing	answers in agreement (number)	answers in agreement (%)
1 expert	39	62%
2 experts	5	8%
3 experts	1	2%
4 experts	3	5%
5 experts	15	24%

Table 3: Inter-expert agreement for ontologies marked as relevant.

that we considered. Swoogle searches only resource labels for the query term. Thus, of the ontologies that we considered, the Swoogle engine would have returned only the ones we considered in the *L* case.

We used the BioPortal search engine to get the data for the *L* and *LV* cases.

5. EXPERIMENT RESULTS

Table 2 shows the number of ontologies that at least one expert identified as relevant for each of the queries. More interesting, Table 3 shows the low level of expert agreement in the ontology relevancy: all 5 experts agreed only on 24% of all answers. 62% of answers had only one expert identify them. The query “anatomy” produced the highest rate of inter-expert agreement, with about half the answers identified by all 5 experts. The query “histology” had the lowest inter-expert agreement: 19 of 21 relevant ontologies were identified only by 1 expert, and the remaining two ontologies were identified by 2 experts.

We used the expert results as the gold standard to determine precision, recall and f-measure (harmonic mean of precision and recall) for the results of our algorithm and the three baseline cases we are comparing it to: *L*, *LV*, and *NULL* (Section 4.2).

Figure 2 and Table 4 show the average precision, recall, and f-measure values for all the cases above.⁶ We show the results for using a different number of pages to create the corpus (2, 5, 10, 20, and 50 pages). We get

⁶Note that the average of f-measure values for the four queries is not equal to the f-measure for the average precision and recall. We show the former value in the table.

	Precision	Recall	F-measure
Query expansion	54%	63%	58%
Null case (<i>NULL</i>)	29%	100%	43%
Labels+values (<i>LV</i>)	65%	27%	40%
Labels only (<i>L</i>)	64%	13%	27%

Table 4: Average values for precision, recall, and f-measure for the 4 queries in different cases, using a 2-document corpus for query expansion.

the best retrieval performance with the smallest corpus, 2 pages, and the f-measure gradually goes down as we take more pages for the corpus. In this case, our average precision is 54% and average recall is 63%, and an average f-measure value for all the queries of 58%.

Note that the retrieval results of looking for the original query terms only in labels (the *L* case) is extremely low: the average f-measure is 27% (precision is 64% and recall is 13%). When we add property values into consideration, the f-measure becomes 40%, which constitutes a 48% improvement over using only labels. This result, however, is slightly worse than simply returning all ontologies from the repository. In the latter case, we get 29% precision and, of course, 100% recall. In other words, in our repository, simply returning all ontologies would have performed better than trying to look for original query terms anywhere. Naturally, this result will not hold with a larger repository, as the precision will go down drastically.

If we compare the f-measure for our ontology-search method based on query-expansion with that of the search for the original query term (*LV*), we get a 43% improvement. Comparing to the traditional ontology search, where the search engine uses only labels and only the original query terms (e.g., Swoogle), our method provides improvement of 113%.

6. DISCUSSION

Building ontologies is difficult and costly, and therefore it is well worth investing in more advanced searching approaches to encourage reuse of existing ontologies. The retrieval performance of current ontology-search techniques clearly shows the need for better searching approaches.

In this paper we described and demonstrated how query expansion techniques can be beneficial in ontology search. This work is inspired by the observation that many queries for searching ontologies name a domain rather than a specific term that must appear among ontology concepts.

In this work we tried to focus on getting the *right* ontologies to stand out above the rest, rather than on their precise ranks.

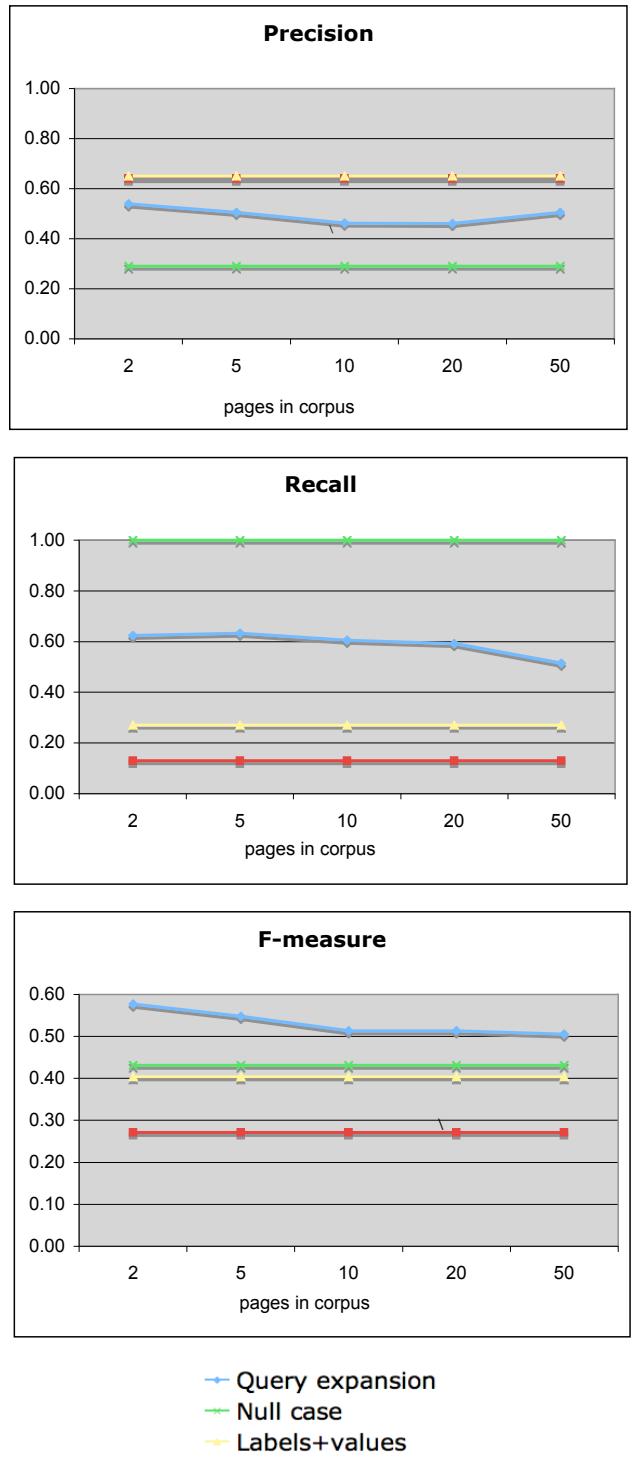


Figure 2: The average precision, recall, and f-measure values for different numbers of pages in the corpus.

The task of ontology search is inherently difficult even for human experts who have access to the complete information. Our results showed a high degree of inter-expert disagreement. There are several possible reasons

for this level of disagreement, mostly, we believe, having to do with the domain (anatomy, physiology, pathology and histology) selected. In general, ontologies in biomedicine were first developed to *name things*; this trend is reflected in the large number of relevant ontologies that cover aspect of body parts, parts of plants, worms and mice [anatomy]. Once the part list is in place, the next step is to talk about things that go wrong in those parts and their interactions [pathology]. Following that, is the dynamics of the various interactions [physiology] and the morphology of the ultra-structure [histology]. The domain of anatomy is well covered by existing biomedical ontologies followed in turn by pathology, physiology, and histology. Therefore, it is not surprising that the query for *anatomy* produced the highest agreement among the experts. With increase in the diversity of biomedical ontologies, and at the same time, with emergence of consensus on their quality, we might expect more agreement among experts on which ontologies cover physiology and histology domains.

How the query is expanded and the result of that expansion will obviously have a significant impact on the final results. If the expanded query contains terms that are too general or irrelevant, then this inaccuracy will propagate to the result set. Therefore, selecting the right corpus is an important first step in our approach. Our results show that using only 2 Wikipedia documents produces better results than using a larger corpus. Indeed, the first page or two from Wikipedia (as returned by the Google search restricted to the Wikipedia site) are dedicated to describing the domain of interest, and hence produce a good set of terms with which to expand the query.

We evaluated our approach against queries on a collection of ontologies in the BioPortal. However, our approach is domain independent and can be used for searching ontologies in other domains. The only requirement is that the query term is reasonably well covered in Wikipedia to allow our system to collect a representative corpus.

In our experiment, some of the TF results contained terms that might be regarded as too general for the domain in question. A better stop-word list can eliminate such terms if encountered. However, if a general term crops in, then it is likely that most of our ontologies will contain that term, and hence it should not affect the final results. Removing the term with the largest number of hits from each ontology score also helps alleviate the effect of spurious or common terms.

Query expansion may not always be required. For example if the submitted query contains many terms, then there is probably no need for much further expansion [11]. However, queries with just a few terms, which

are very common, will most likely always need some sort of expansion when searching for ontologies. Of course there can be different ontology searching needs, such as searching for a specific triple or class structure. Such requests might also require some sort of expansion, for example to find similar or imprecisely matching triples, or more elaborated class structures. Distinguishing whether a query is for the specific terms or whether the user meant for the terms to be treated as the ontology topic or domain will help provide relevant results. It is not clear how to make this determination without asking the users to go through the extra step of specifying what they are looking for

If metadata for ontologies were widely available, then many ontologies would have the domain they cover as part of the metadata. However, in order to enable ontology-search engines to know which metadata fields to search, this metadata description must follow some commonly agreed standard (use a common *metadata ontology*). If both the metadata standards and the metadata descriptions were widely available, the search could be limited to the content of this metadata, rather than the content of the ontology itself. However, even in that case there is the danger that not every ontology for which its metadata say that it is “about anatomy” is indeed about this domain, or covers the domain well enough to be considered a *relevant* ontology. Google has long stopped using Web page metadata (e.g. keywords) when searching as they often get misused by some Web page authors to mislead the search engine, and misdirect the end user to their pages. The only way to verify the claims of an ontology’s metadata and to find out what it is *really* about, is to perform further analysis on the content or structure of the ontology itself.

We reported our results only for one of the set of parameters with which we experimented (naturally, the one that consistently produced the best results). We have also considered different ways of deciding which ontologies to return given their score. Returning a fixed number of ontologies (say, top 10 or top 20) is not a good option: for queries with many relevant ontologies (e.g., “anatomy” in our repository) such approach is likely to have a low recall; for queries with few relevant ontologies (e.g., “pathology” in our setting) such approach will have low precision. Thus we chose to return the ontologies with scores above the median of all non-zero scores. This approach takes into account the query itself.

7. SUMMARY AND FUTURE WORK

When performing the experiments to evaluate the efficacy of our method for ontology search, we have discovered several unexpected facts related to the problem. First, inter-expert agreement in determining ontologies relevant to a user query is extremely low: only in 24% of the cases all 5 of our experts agreed on an answer.

Second, using only the query term and searching only labels of classes and properties provides extremely poor retrieval performance (f-measure of 27%). Using property values in the search in addition to labels improves the results by 48%. Using the query-expansion method that we discussed and searching in labels and property values improves the result by 43%.

Despite the improvement in the quality of results, we envision several avenues that we can pursue to improve the efficiency of ontology search further.

First, we plan to study other approaches for finding a corpus that relates to a given domain. For example, we can expand the query using a thesaurus, such as UMLS or WordNet before searching the Web for related documents to increase the chances of finding highly relevant Web pages.

Second, instead of using TF analysis to find terms with which to expand a query, we plan to investigate other approaches. For example, we can look at the hyper-linked terms in the corpus pages: these terms may be the most relevant ones in the pages. We can also consider frequent term co-occurrences and term combinations in the corpus.

Third, as we mentioned earlier, our approach addresses only the problem of ontology search. We envision combining our results with various ontology-ranking methods to help users understand which of the returned ontologies are most relevant to their query. The holy grail of the ontology search lies in providing highly relevant ontologies with the best results shown first. And that's the goal we are striving to achieve.

Acknowledgement

This work is partially supported by the Leverhulme Trust. It was also supported by the National Center for Biomedical Ontology, under roadmap-initiative grant U54 HG004028 from the National Institutes of Health. We are especially grateful to Mathew Jones for implementing some of the code used in this system, and to Daniel Rubin in helping to collect results from experts. Special thanks to the researchers in biomedical ontologies who provided expert results.

8. REFERENCES

- [1] H. Alani and C. Brewster. Ontology ranking based on the analysis of concept structures. In *3rd Int. Conf. Knowledge Capture (K-Cap)*, pages 51–58, Banff, Canada, 2005.
- [2] H. Alani, C. Brewster, and N. Shadbolt. Ranking ontologies with aktiverank. In *5th Int. Semantic Web Conf. (ISWC)*, Athens, Georgia, USA, 2006.
- [3] C. Brewster, H. Alani, S. Dasmahapatra, and Y. Wilks. Data driven ontology evaluation. In *Int. Conf. on Language Resources and Evaluation*, Lisbon, Portugal, 2004.
- [4] P. Buitelaar, T. Eigner, and T. Declerck. Ontoselect: A dynamic ontology library with support for ontology selection. In *Proc. of the Demo Session at the International Semantic Web Conference*, Hiroshima, Japan, 2004.
- [5] K. Collins-Thompson and J. Callan. Query expansion using random walk models. In *ACM Fourteenth Conf. on Information and Knowledge Management (CIKM)*, Germany, 2005.
- [6] L. Ding, T. Finin, A. Joshi, R. Pan, R. S. Cost, Y. Peng, P. Reddivari, V. C. Doshi, and J. Sachs. Swoogle: A semantic web search and metadata engine. In *Proc. 13th ACM Conf. on Information and Knowledge Management*, Nov. 2004.
- [7] S. Harris and N. Gibbins. 3store: Efficient bulk rdf storage. In *Proc. 1st Int. Workshop on Practical and Scalable Semantic Systems (PSSS'03)*, pages 1–20, Sanibel Island, FL, USA, 2003.
- [8] H. Lewen, K. Supekar, N. F. Noy, and M. A. Musen. Topic-specific trust and open rating systems: An approach for ontology evaluation. In *Proc. of the 4th EON workshop, 15th Int. World Wide Web Conference*, Edinburgh, 2006.
- [9] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford University, 1999.
- [10] J. Z. Pan, E. Thomas, and D. Sleeman. Ontosearch2: Searching and querying web ontologies. In *Proc. IADIS Int. Conf. WWW/Internet 2006*, pages 211–219, University of Murcia, 2006.
- [11] 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*, 11:95–130, 1999.
- [12] C. Rosse and J. L. V. Mejino. A reference ontology for bioinformatics: The foundational model of anatomy. *Journal of Biomedical Informatics.*, 2004.
- [13] M. Sabou, V. Lopez, and E. Motta. Ontology selection on the real semantic web: How to cover the queens birthday dinner? In *Proc. 15th Int. Conf. on Knowledge Engineering and Knowledge Management (EKAW)*, Czech Republic, 2006.
- [14] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw- Hill, 1983.

- [15] E. M. Voorhees. Query expansion using lexical-semantic relations. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 61–69. Springer-Verlag New York, Inc., 1994.