An ontology-driven approach to Web search: analysis of its sensitivity to ontology quality and search tasks

Stein L. Tomassen
IDI, NTNU
Sem Saelandsvei 7-9,
NO-7491 Trondheim, Norway
+ 47 735 94218
stein.l.tomassen@idi.ntnu.no

Darijus Strasunskas
IOT, NTNU
Alfred Getz veg 3,
NO-7491 Trondheim, Norway
+ 47 735 93659
darijuss@gmail.com

ABSTRACT

An increasing number of recent information retrieval systems makes use of ontologies to help the users to detail queries and to come up with semantic representations of documents. A particular concern here is user-friendliness (usability) and scalability of those approaches for Web search purposes. In this paper, we present an approach where entities in an ontology are associated with domain terminology by feature vectors (FV). A FV reflects the semantic and linguistic neighbourhoods of a particular entity. The semantic neighbourhood is derived from an ontology and is based on related entities and specified properties, while linguistic neighbourhood is based on co-location of terms in a text corpus. Later, during the search process the FVs are used to filter and re-rank the search results of the underlying search engine and thereby increasing the precision of the result.

We elaborate on the approach and describe how the FVs are constructed. Then we report on a conducted evaluation where we analyse the sensitivity of the approach w.r.t. ontology quality and search tasks. Results indicate that the proposed approach and implemented prototype are able to improve the search results of a standard Web search engine. Furthermore, the analysis of the experiment data shows that the level of ontology specification is important for the quality of the FVs.

Categories and Subject Descriptors

H.3.3 [**Information Storage And Retrieval**]: Information Search and Retrieval - *information filtering, selection process.*

H.3.4 [**Information Storage And Retrieval**]: Systems and Software - *performance evaluation (efficiency and effectiveness).*

Keywords

ontology, web search, semantic search, ontology quality, feature vector construction.

1. INTRODUCTION

Given broad Web terminology and limited domain terminology used in an ontology, we endeavour to semantically and linguistically extend domain terminology (terms used to name entities in a domain ontology) in order to improve matching between ontology entities and terminology of documents. The approach presented in this paper utilizes ontologies that are automatically adapted to the corpus' terminology by computing a feature vector (FV) for each entity in the ontology. The idea is to associate every entity (classes and instances) with a FV to tailor these entities to the specific terminology used in the text corpus (the Web). Synonyms and conjugations naturally go into such a vector, but we would also like to include related terms tended used in connection with the entity and provide a contextual definition of it. The FVs are later used to filter and re-rank the search results from an underlying search engine before presentation of the final result. We envision our approach to be used in transition from the current Web and the fully-fledged Semantic Web.

Web search is characterized by having focus on retrieving documents, navigating to a particular Web page, or retrieving a piece of wanted information rather than browsing knowledge or answering a question. Employing ontologies to enhance this type of searches requires certain qualities of the ontologies. For instance, subclass hierarchies are considered sufficient for document retrieval while any other ontology specifications (properties and axioms) are required only for knowledge browsing and question answering [10]. However, here we show that ontology quality improvement, by specifying equivalent and disjoint classes, adding instances, and properties, can significantly improve Web search results.

The objective of this paper is to present the proposed approach, analyse and discuss the results from an experiment. The experiment has been conducted with potential end-users of such systems. The approach and prototype are evaluated by the means of an experiment and a post-task questionnaire. The paper addresses broad evaluation research questions as follows.

RQ1. How sensitive is the approach to ontology quality? Feature vectors are built based on knowledge specified in ontologies, therefore granularity and quality of encoded knowledge has direct impact on the quality of FVs.

RQ2. Is the approach performance indifferent to various search tasks? Various search strategies and information needs typically concern different granularity of required information. For instance, some prefer finding concrete and concise information on a particular topic, while some are interested in exploring a topic either in depth or in breadth.

Consequently, the novelty and contribution of this paper lies in the analytical experiment attempting to deepen understanding how ontology quality and search tasks aspects affect the overall performance of semantic search.

The rest of the paper is structured as follows. First, we briefly review related work. Then we elaborate on the proposed approach to ontology-driven Web search. Next, we describe a conducted experiment where we evaluate the proposed approach and its sensitivity to ontology quality and search tasks. Then the main results are presented followed by a detailed analysis and discussion. Finally, we conclude the paper and outline future work.

2. RELATED WORK

The Web contains vast resources of information. However, the diversity of topics and terminologies makes it difficult to find relevant information. The Semantic Web (SW) is believed to be the successor of the current Web and provides means to tackle some of these issues. The grand idea is to annotate every piece of information with machine-processable semantic descriptions that enable more advanced usage of the information elements, like reasoning among others. Consequently, there are many initiatives to semantic search. Some are relying on semantic annotations (e.g., [27]); some are enhancing clustering of retrieved documents (e.g., [17]). There are also many efforts devoted to research on improvement of information retrieval (IR) by using SW techniques. Most of these approaches are utilizing ontologies with encoded domain knowledge to improve search (e.g., [2, 4, 21, 26]). In this section section, we will explore related work where SW techniques are used to enhance search. Since we focus on search task fitness in this paper, a brief overview of information needs and search strategies are provided at the end of this section.

2.1 Semantic search

Search systems for the SW can generally be divided into two categories; those searching for SW documents (i.e., documents expressed in a semantic mark-up languages like OWL, RDF, etc) and those using SW techniques to improve search results [7]. The overview provided here is limited to approaches that endeavour improvement of search by SW techniques (for a more extensive overview of SW systems the reader is referred to [7, 13, 20]). Next, we will provide an overview of the most similar approaches to our work.

Many approaches typically enhance traditional vector space model (VSM) by adding processing of semantics. Nagypal [15] combines ontology usage with the VSM by extending a non-ontological query. There, ontology is used to disambiguate queries. Text search is run on the concepts' labels and users are asked to choose the proper term interpretation. Paralic & Kostial [18] describe a similar approach where documents are associated with concepts in the ontology. The concepts in the query are matched to the concepts of the ontology in order to retrieve terms and then used for calculation of document similarity.

OntoSearch by Jiang & Tan [12] is a full text search engine that depends on documents annotated with elements from an ontology. The user submits a traditional keyword-based query that yields a set of documents. These retrieved documents contain semantic annotations that are used by the spreading activation algorithm to retrieve additional documents and finally rank the documents. Results show that the approach performs better than a compared keyword-based approach.

Formica et al. in [8] proposes a novel way of ranking annotated documents with respect to both an ontology and a user query. In advance, the documents have been annotated with a set of characterizing concepts, called a feature vector, which they assume already have been built. These FVs function as instances of the corresponding concepts. Similarity between the concepts of a user query and the FVs with respect to the ontology are calculated. Testing shows that their approach performs slightly better than other compared approaches. However, a limitation with the approach is that only the hierarchical structure of the ontology is used when calculating the similarity scores.

The approach by Solskinnsbakk & Gulla [22] is relying on constructing ontological profiles that contain concept vectors. However, when creating the concept vectors they are depended on a highly relevant document collection. Furthermore, they also need a collection of non-relevant documents in order to construct negative concept vectors. Both vectors are used in query expansion. Testing shows good results for situations where recall is more critical then precision.

2.2 Information needs and search strategies

There are many studies of users' information needs, their search strategies and behaviour (e.g. [1, 11]) resulting in different classification of search strategies. For instance, Guha et al. [9] distinguish two different kinds of search, namely, navigational search and research search. Navigational search is defined as the one where the user provides a phrase or keywords and expects to find them in the documents, i.e. the user is using a search engine to navigate to a particular document. While in the research search the user provides a phrase or keywords that are intended to denote object or phenomena about which the user wants to gather information, i.e. the user is trying to locate a collection of documents which will provide required information [9].

With the emerging Semantic Web there is envisioned a shift in IR from retrieval of appropriate Web pages to answering questions without extraneous information [14]. This, being separate and important areas in information retrieval and knowledge management, requires robust ontology quality, reasoning, and fine-grained annotation of documents. However, precise question answering is the most ambitious information retrieval task but still inevitable and a required feature of Web search. Therefore, we consider a fact-finding search being able to partially substitute question answering on the Web. For this reason, we adopt a classification of search tasks into the following categories: fact-finding, exploratory, and comprehensive search tasks [1]. In fact-finding, a precise set of results is important, while the amount of retrieved documents is less important. In exploratory search task, the user wants to obtain a general understanding about the search topic, consequently, high precision of the result set is not necessarily the most important thing, nor is high level of recall [1]. Finally, a concern of comprehensive search task is to find as many documents as possible on a given topic, therefore the recall and precision should be as high as possible.

3. ONTOLOGY-DRIVEN SEARCH

In this section, we elaborate on our approach. We start with an introduction to feature vectors then describe the process to construct FVs and finally finish the section by describing how FVs are used in search.

3.1 Introduction to feature vectors

The development of the approach is inspired by a linguistics method for describing the meaning of objects - the semiotic triangle [16]. In our approach, a feature vector "connects" a concept (entity) to a document collection, i.e. a FV is tailored to the specific terminology used in a particular document collection (see Figure 1). FVs are built considering both the semantics encoded in an ontology and the dominant lexical terminology surrounding the entities in a text corpus. Therefore, a FV constitutes a rich representation of the entities and is related to the actual terminology used in the text corpus. For a more formal definition of a FV, the keen reader is referred to [24].

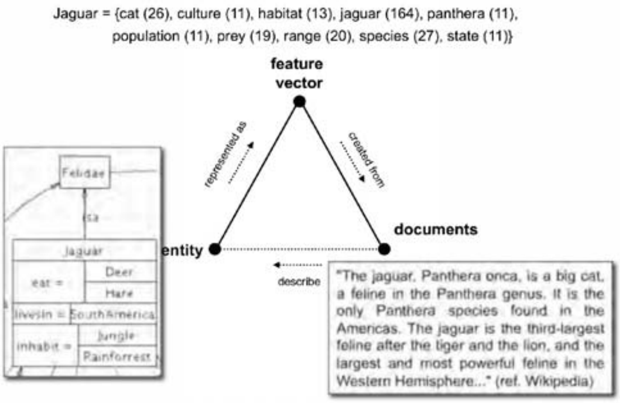


Figure 1. Explanation of a FV by adapted semiotic triangle. In addition, an illustration of a *feature vector* created for the *entity* `Jaguar` with an ontology fragment (Animals[2]) depicting the `Jaguar` *entity* together with a text fragment (*documents*) from the Web being related to the entity.

The process of selecting relevant entities and terms (words) into these sets is elaborated in the Section 3.3, but first the overall architecture of the approach is presented in the next subsection.

3.2 Architecture

Figure 2 depicts the overall architecture of the ontology-driven information retrieval system. In this section, we will briefly describe the architecture and its components (more details are provided in [25]).

The system consists of both offline and online components (with respect to actual search process). The offline components are used to add and populate new ontologies (Section 3.3) while the online components use the already populated ontologies in search (Section 3.4). The underlying query and indexing system is used both offline and online.

3.3 Feature vector miner

The feature vectors are composed from both the semantics encoded in the ontologies and the surrounding terminology of the entities in a text corpus (the Web). A simplified version of the FV construction process is depicted in Figure 3 (more details can be found in [24]). The process of constructing FVs constitutes main phases.

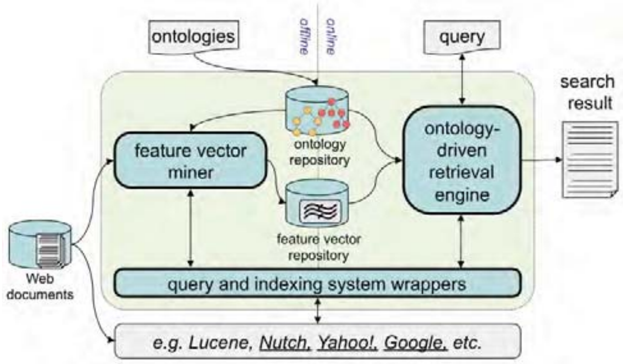


Figure 2. An overview of the ontology-driven information retrieval system and its components.

The main aim of the first phase is to extract and group sets of candidate terms being relevant to each entity. First, an ontology is analysed to find the entities and the relationships among them. Then a query for each entity is composed. The queries are constructed using the entity name and expanded with neighbouring entities (i.e. *parent*, *child*, and/or *other* [24]). The queries are submitted to the underlying search system. The result of this is a set of retrieved documents for each entity. Each document set is clustered to group documents having high similarity. For each cluster a set of candidate key-phrases, noun phrases collocated with the entity, are extracted from the documents of the cluster. These sets candidate key-phrases (represented as a Cluster Feature Vector (CLFV)) associated with each entity are the input to the next and final phase of the process namely identifying and creating the final FVs of the entities.

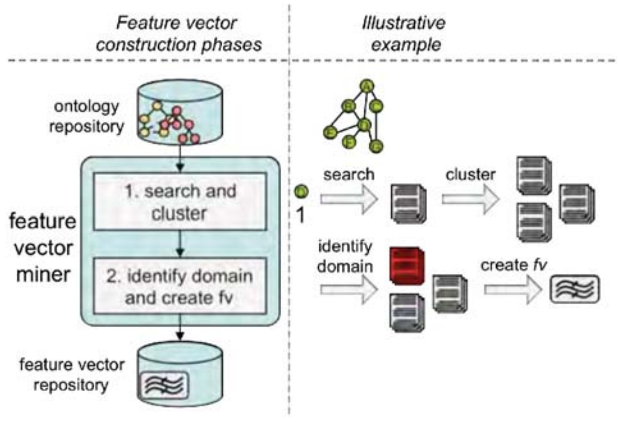


Figure 3. On the left hand side, a simplified version of the Feature Vector Construction process is depicted, while an illustrative example is found on the right hand side.

At this stage of the process, we do not know which of the clusters (CLFVs) for each entity are most relevant to the domain of interest defined by the ontology (e.g. the concept of Jaguar can be part of many different domains like being a car brand, animal, operating system, etc.). Consequently, the main aim of the last phase is to identify the most relevant clusters w.r.t. the entities defined by the ontology. The hypothesis is that individual clusters having high similarity across ontology entities are with high probability of the same domain (e.g. *Jaguar* w.r.t. *Felidae* depicted in Figure 1). This hypothesis is backed up with observed patterns of collocated terms within the same domain, and consequently different domains will have different collocation pattern of terms. However, the similarity of clusters depends a lot on the quality of the ontology, especially on the semantic distance between the entities. Therefore, the most prominent cluster is found by calculating the similarity between the CLFVs of the current entity with all the CLFVs of the neighbouring entities. Then finally the clusters with the highest score are selected and used to create the FVs for each entity. The result of this process is a FV for each entity with key-phrases that are associated with both the entities and the domain defined by the ontology.

3.4 Ontology-driven retrieval engine

In this section, we will describe the ontology-driven search engine where feature vectors are used to disambiguate search.

First, the user needs to formulate a query. The user can specify one or more entities related to the domain of interest (if no entities are specified then ordinary keyword search is performed). In addition, the user can specify a set of keywords to narrow the search even further (see Figure 6). By differentiating on entities and keywords, the real intention of the user's query can better be interpreted by the underlying machinery and thus present more relevant results.

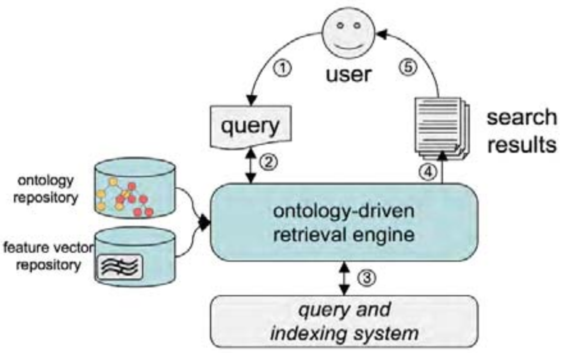


Figure 4. An overview of the search process.

Figure 4 depicts an overview of the different steps of the search process. Firstly, the user initializes a search (1) by submitting a query $Q=\{e_1, \ldots, e_n, k_1, \ldots, k_m\}$, where e is an entity part of an ontology O, k is a keyword, n is the total number of entities while m is the total number of keywords, to the *ontology-driven retrieval engine* (2). Then, the retrieval engine identifies the corresponding entities of the ontologies and submits a semantically enriched query $Q'=\{S_{e1}, \ldots, S_{en}, k_1, \ldots, k_m\}$, where S_e is a semantically enriched (i.e. the entity name and selected neighbouring entities) entity e, to the underlying *query and indexing system* (3). Those query terms with no corresponding entity are treated as ordinary keywords. For each document of the search result, a document feature vector (DFV) is created (3). Then the search result from the underlying search engine is filtered and re-ranked by comparing the similarity between the FV_es of Q and the DFVs. Only those documents having a similarity score with the FV_cs of Q above a certain threshold are selected and next ranked according to the similarity scores (4) before presented to the user (5).

4. DESIGN OF EXPERIMENTS

In this section, we present the experiments conducted to evaluate the approach with respect to its sensitivity of both ontology quality and search task.

Table 1. Demographic information about the participants.

Demographic feature	Response	Demographic feature	Response	Demographic feature	Response
Gender male:	18 (86%)	**Amount of keywords in a good query**		**Knowledge about ontologies**	
female:	3 (14%)	2 or less	4 (19%)	None	1 (5%)
		3	11 (52%)	Have heard about	9 (43%)
Age [18-24]:	13 (62%)	4	6 (29%)	Have been studying	5 (24%)
[25-29]:	5 (24%)	5	0 (0%)	Have been using in prototyping	6 (29%)
[30-39]:	2 (9%)	6 or more	0 (0%)	Practical development	0 (0%)
[40-49]:	1 (5%)				
Web search experience		**Search service preference**		**Participation in evaluations**	
None:	0 (0%)	Generic Web search:	20 (95%)	First time:	4 (19%)
Sparse:	0 (0%)	Specialized Web search:	5 (24%)	Sparse:	7 (33%)
Moderate:	5 (24%)	On-line catalogues:	0 (0%)	Moderate:	8 (38%)
Extensive as user:	10 (48%)	Specialized digital libraries:	8 (38%)	Extensive as participant:	1 (5%)
Extensive as user and developer:	6 (28%)	Other (journal site, wikipedia, google specialised search):	3 (14%)	Both as participant & evaluator:	1 (5%)

4.1 Experiment settings

The participants in our experiment were mainly 4th year students at the Norwegian University of Science and Technology (NTNU) (see Table 1 for demographic information about the participants). 21 subjects participated that were offered payment for used time after full completion of the experiment.

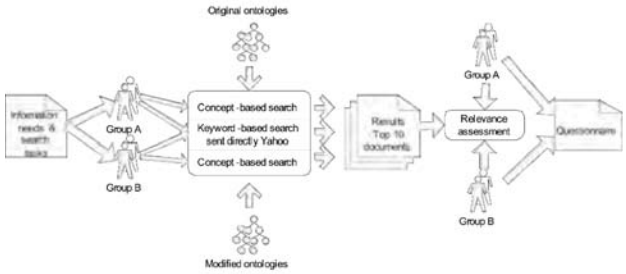


Figure 5. Design of the experiment.

Setting for the experiment is elaborated in Figure 5. The participants of the experiment were given eight topics and descriptions of information needs from four different domains. Queries were specified using keywords and entities from particular domain ontology. They needed to formulate in total 16 queries each (eight submitted to the prototype and eight to the baseline). The system returned 10 top ranked documents for each query, which they assessed based on their relevance to the participants perception of the topic description. After finishing the experiment, the participants completed a questionnaire of 29 questions.

The simulated situation tasks were as follows:

Food & Wine domain

1. *Explorative search task*. What grapes are used to make suitable wines to beef curry[1].
2. *Fact-Finding search task*. Find a perfect dessert wine for a dessert made from chocolate with sweet fruits.

[1] Actually, the users were given a more detailed and verbose description of the topics and information needs in order to define them precisely and avoid ambiguities.

Travel domain

3. *Comprehensive search task*. Try to get an overview of the kind of safaris that are available.
4. *Fact-Finding search task*. Find out the possibilities for a leopard safari.

Animal domain

5. *Explorative search task*. Explore facts about jaguars with the purpose of writing an essay.
6. *Comprehensive search task*. Survey information about jaguars, leopards, and other members of the cat family.

Autos domain

7. *Fact-Finding search task*. Find out about the car brand named Saturn.
8. *Comprehensive search task*. Get an overview of SUVs.

The participants were divided into two groups that used different ontologies[2] for the same domain (see Figure 5). The first group used the original ontology while the second group used an altered version of the original ontology. The original ontology was altered to include more relations and/or instances to see if this would influence on the search results. All four ontologies were modified by adding instances (all ontologies), specifying additional object properties (travel, animal, and wine ontologies) and refining taxonomical relationships (animal ontology). The results of these changes were different feature vectors generated for the same entities of the two different but still similar ontologies (see Table 2). In summary, group 1 contained 10 participants, while group 2 had 11 participants. In total, the users executed 81 queries using the original ontologies and 92 queries using the modified ontologies, and 152 were simple keyword based queries executed directly to the baseline.

Table 2. Ontology and FV characteristics.

Domain	Ontology version	Ontology characteristics			Feature vectors' characteristics	
		# of concepts	# of instances	# of properties	avgerage length	avgerage cosine similarity
Food & Wine	1	82	155	14	36,66	0,92
	2	83	157	17	38,38	
Travel	1	34	14	10	34,67	0,92
	2	34	29	10	37,26	
Animal	1	51	0	2	33,04	0,78
	2	63	15	8	36,12	
Autos	1	90	321	16	33,27	0,87
	2	91	328	16	33,65	

We choose to use the Yahoo! Web Search API as the backend search engine that consequently also performed as our baseline for our comparison. Ideally, we would use the Text Retrieval Conference (TREC)[3] data as baseline. However, we experienced the same problems as d'Aquin et al. [6] in finding good ontologies that covered TREC. In fact, d'Aquin et al. [6] found that those ontologies available on the Web covered only 20 percent of the domains described in TREC (they used the 100 queries from the WT10G test collection). As a result, we choose to use Yahoo! Web Search as baseline and let the participants do a qualitative perceived relevancy of the top 10 results.

We adopted the query scoring and calculation method presented by Brasethvik [3] to measure the qualitative perceived relevancy. The participants needed to mark (as either *trash*, *irrelevant* or *duplicate*, *related*, or *good*) each of top 10 retrieved documents according to perceived relevance. The final relevance score for a query falls into a range [-50, 100] (more details are provided in [23]). The relevance score substitutes a conventional precision metric. We have decided to focus on precision instead of recall since we aimed at improving Web search results, where precision (i.e. relevant documents at top positions) is more important.

4.2 Prototype implementation

A prototype was implemented in Java and the experiments were performed on a standard PC with an Intel™ Pentium processor running Windows™ XP, running Apache Tomcat. A Web user interface similar to a typical search engine found on the Web was created (Figure 6) to make the interface as familiar as possible to the user. To assist the user in finding appropriate entities of the ontology a suggest-like interface was implemented (i.e. when the user started to type a list of suggested entity names were provided that the user could select from).

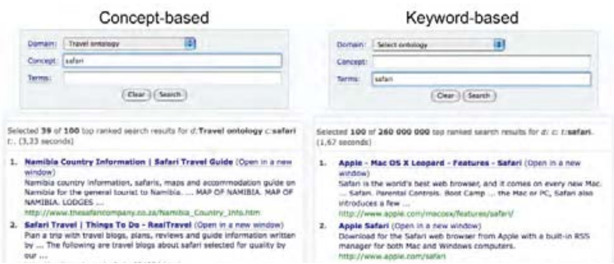


Figure 6. The search user interface of the prototype, concept-search vs. keywords-based.

The implemented prototype was configured to use the Yahoo! Web Search API[4] as the backend search engine that also performed as the baseline for our evaluation.

4.3 Ontology quality assessment - the EvOQS framework

In this subsection we briefly overview the EvOQS (Evaluation of Ontology Quality for Searching) framework [23] for evaluation of ontology quality in search applications. A part of this framework has been used to assess ontology quality in the experiment. Here we will focus on that part of the framework, for a complete framework description the keen reader is referred to [23].

The framework defines a stepwise ontology selection procedure and metrics. Ontology quality aspects are defined with respect to the search tasks and search enhancement requirements. The framework adopts earlier discussed (Section 2.2) classification of search tasks into three categories, such as *fact-finding*, *exploratory*, and *comprehensive* search tasks [1]. In this paper, we focus on *search task fitness*. This step concerns evaluation of ontology fitness for a particular search task. For instance, ratio of taxonomic vs. non-taxonomic relationships is important when selecting an appropriate ontology for exploratory and comprehensive search tasks. For instance, in *fact-finding,* a high precision can be achieved by using precise terms or phrases in the

[2] The ontologies used are all formalized in OWL and can be found here: http://research.idi.ntnu.no/IIP/ontologies/.

[3] Text Retrieval Conference (TREC), http://trec.nist.gov

[4] Yahoo! Developer Network, http://developer.yahoo.com

query, typically by formulating a query consisting of several terms. In order to enhance results in fact-finding search task provided entities needs to be extended by their instances and datatype properties. Consequently, entities, their instances and properties, are essential here. In *exploratory search,* the user may find topic-related documents by extending simple keyword-based search with parent- and child-entities. In order to cover broader-topics in *comprehensive search,* hypernyms and hyponyms, sibling entities, and semantic relationships are in addition included in the query to cover the most important aspects of the search topic.

The ontology elements that are necessary to support search tasks can be summarized as follows. We compute fact-finding fitness of an ontology as a combined proportion of specified instances and properties vs. specified classes, while explorative fitness measure is based on an average amount of subclasses specified for a class in an ontology or entity cluster. Finally, a metric for comprehensive search fitness is calculated as fraction of object properties, super-classes, sub-classes, and sibling-classes w.r.t. the total amount of entities. Recall that all four ontologies were modified by adding instances, specifying additional object properties, and refining taxonomical relationships. The results of these modifications were different FVs created for the equivalent entities.

5. RESULTS ANALYSIS

In this section we analyze the data collected during experiment described in Section 4. We begin with a generic analysis of the system performance (more information about this generic evaluation can be found in [23]), and then we look at how the modifications of the ontologies changed the feature vectors. Finally, we analyze the sensitivity of the approach to ontology quality and search tasks.

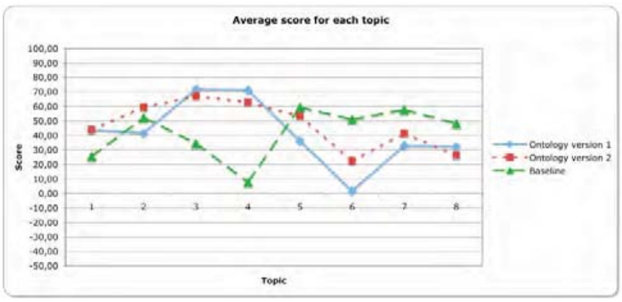


Figure 7. The average scores for each of the eight information-needs described in Section 4.1.

Figure 7 depicts a graph showing how the different ontologies versions influence on the search result relevance score. Recall that ontology version 2 is an altered edition of version 1 having different granularity and level of knowledge specification. The graph shows in general that a more advanced ontology in the sense of having more relations, properties, and individuals does perform better than a similar simpler ontology (see also Table 4).

From Table 3 we can explain the biggest improvement vs. the baseline in topics 3&4 (see Figure 7 and highest scores for ontology usefulness in 3rd column of Table 3). Topic 6 scored lowest on the description and presence of concepts in description (that means it is more difficult to formulate query as consequence), as well topic familiarity and ontology usefulness received third lowest rates – obvious in Figure 7. Furthermore, topic 6 had biggest variance in query length.

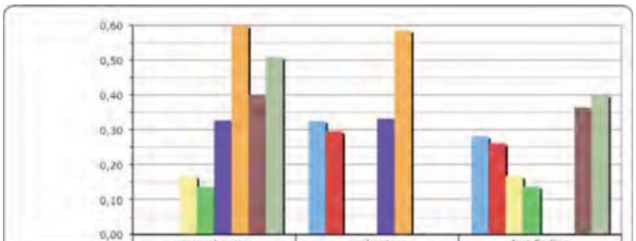
Table 3. Mean scores on questionnaire items regarding the experiment. Answers were measured using Likert 5-point scale (from lowest to highest relevance, familiarity, etc.)

Topics	Familiarity w/retrieval tasks	Ontology usefulness	Quality of info needs and task descriptions	Presence of concepts in descriptions
1	**2,43**	3,48	3,81	2,67
2	**2,33**	3,43	3,86	**2,43**
3	2,62	*3,57*	*4,10*	*2,86*
4	2,62	*3,76*	3,95	2,62
5	*2,76*	3,38	3,90	2,67
6	2,71	3,14	**3,71**	**2,38**
7	2,57	**2,81**	*4,05*	*2,86*
8	*2,86*	**2,95**	**3,71**	2,71

Remark: Lowest values are in bold, while highest in italic.

When observing the length of the queries, it also seems to be a trend that the prototype performs better for shorter queries compared to keyword-based queries which is also observed for other entity-based approaches (e.g. [5]). The entity-based queries were also in general shorter than keyword-based queries.

Another observed pattern is how the users formulate their queries. Recall that the groups were divided into sub-groups. The first group needed to formulate the keyword-based queries prior to the entity-based queries and the other sub-group vice versa. The group formulating the entity-based queries first did in average use 13% less keywords and 14% fewer entities compared to the group formulating the entity-based queries last. Note that a query must contain one or more entities in combination with zero or more keywords to be classified as an entity-based query. However, the group formulating the keyword-based queries first had a tendency to use most of the keywords in the entity-based search as well, consequently having in general longer entity-based queries than the other group. The keyword-based queries for both groups were almost equal in length with a difference of only 2%.



	comprehensive	exploratory	fact finding
Wine ont. ver. 1		0,32	0,28
Wine ont. ver. 2		0,30	0,26
Travel ont. ver. 1	0,17		0,17
Travel ont. ver. 2	0,14		0,14
Animals ont. ver. 1	0,33	0,33	
Animals ont. ver. 2	0,60	0,59	
Autos ont. ver. 1	0,40		0,36
Autos ont. ver. 2	0,51		0,40

Figure 8. The average neighbouring FV similarity scores of those entities used in the experiment w.r.t. search task.

5.1 Analysis of FV quality and its impact

Figure 8 depicts a graph showing how the average neighbourhood similarity score differs with respect to the search task. In this overview, only those entities that were used in the experiment were considered. Similarity was measured by a standard cosine similarity measure. The graph shows that for the Travel and Wine ontologies the similarity decreases from ontology version 1 to 2 while it is opposite for the Animals and Autos ontologies. High similarity indicates that an entity's FV is fairly equal to its neighbouring entities' FVs, while a low similarity indicates a more unique FV with respect to its neighbours. We can also observe that the Travel ontology has the lowest neighbourhood similarity scores compared with the other ontologies, but also had the highest relevance scores found in Figure 7 (topic 3 and 4). This indicates that more unique FVs are beneficial versus more general FVs. Therefore, we can assume that the changes done with the Animals ontology, from version 1 to 2, had a negative effect on the uniqueness of its FVs since the neighbourhood similarity increased considerably, but still version 2 performed better than version 1 (see Figure 7 topic 5 and 6).

Table 4. Comparison of mean relevance score of keyword and entity based searches

	Mean relevance score	Diff. from baseline
Keyword-based	42.2	-
Ontology ver. 1	42.1	-0.2%
Ontology ver. 2	46.6	10.5%

From Table 4 we can observe that the modified ontologies significantly increased performance of the prototype. The improvement resulted to be more than 10 percent. Given such significant enhancement we take a closer look at the ontology quality and the role of search tasks in the next subsection.

5.2 Ontology quality impact and search task performance

As a result of the earlier discussed modification of the ontologies, comparing the relevance scores for the original ontologies vs. the modified ones, we found an improvement in mean score that equals to 10.5% (the overall mean relevance for original ontologies score was 42.1 vs. 46.6 for modified ontologies). See Table 4 for comparison of mean relevance scores and Figure 7 for comparison per search topic.

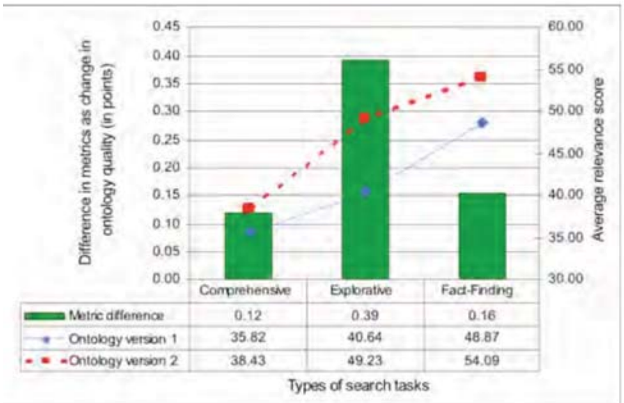


Figure 9. Comparison of ontology quality and search performance based on search tasks [5].

[5] Difference of metrics was calculated using equations verbally described in Section 4.3 and formally defined in [23] and metric value of ontology version 1 has been subtracted from the metric value of ontology version 2 (for corresponding search tasks).

The difference in ontology fitness metrics (see Figure 9) well explains corresponding improvement in performance for the analysed search tasks (r^2=0.905). Most significant improvement has been observed in explorative search task, second largest enhancement resulted in fact-finding search task. Addition of more instances and object properties improved the mean relevance score of fact-finding search tasks, while the addition of sub-classes resulted in better performance of exploratory and comprehensive search tasks.

5.3 Threats to validity

We devote this subsection to discuss possible threats to the results already presented above.

− External validity describes a degree to which the results can be generalized outside the experiment. The experiment has been conducted using only one system (the prototype implementation of the outlined approach), however the conclusions and lessons learnt are applicable to all similar approaches, especially ones using ontology, but limiting its usage to sub-class relationships only.

– The case study is executed at the university. However, the experimental tasks and information needs were chosen from the "real world" and most of the test subjects had extensive search experience (16 out of 21 identified having extensive search experience as search users, six of them identified themselves as having a developing experience in addition to extensive search experience). However, we believe that results would be much more in favour of the proposed system if users that are more diverse were selected (subjects were mostly Computer Science students having a bit more sophisticated skills).

− Users provided subjective evaluations. The individuals needed to interpret the experimental materials and tasks according to their experience. The intention was to create an experiment similar to the real usage of Web search, where users formulate and assess relevance by themselves. Experience seemed to be similar for most of the individuals. However, we observed a difference (variance) among users' queries and document relevance judgments.

− Fatigue effect. On average 2.5 hours were spent to complete the tasks and fill the questionnaire. Therefore, this effect is not considered relevant.

5.4 Concluding discussion

The participants were in general satisfied with the relevance of the results and the prototype performance. They also found the approach particularly helpful in formulating queries for unfamiliar domains. Analysis of the experiment results shows that users tended to formulate shorter queries for the entity-based approach versus the traditional keyword-based approach. This indicates that they have a prior expectation of such a system compensating the lack of provided information in one way or another.

Furthermore, in the survey, the participants were asked to rate the quality of the results compared to the base system in a scale from 1 (very bad) to 5 (very good), and the mean score was 3.5. This score indicates that the approach for automatic construction of entity feature vectors based on any ontology works quite well and its implementation was not bad either, i.e. the users liked "simplicity" of the ontology-driven search interface.

In summary, we have shown that the proposed approach and its preliminary implementation are apt to improve search performance. However, performance of the approach is dependent on ontology quality (level of knowledge specification). While analyzing the results and trying to find an answer to RQ1, we found difference of 10.5% in improved performance due to enhanced quality of ontology. These findings call for further research on how to tailor FV construction to various search tasks (however, this may require more complicated interface) and research to try different techniques in order to reduce sensitivity of the approach to quality of ontology. Analysis of different search tasks and corresponding performance of the approach on those tasks (RQ2) has shown that certain ontology elements have bigger effect on certain information tasks than other ontology elements (recall Table 5). Furthermore, the approach has shown the best performance in the fact-finding category of search tasks, having almost 50% higher relevance score if compared to the comprehensive search task (Table 5). This indicates a need for further research on tailoring the approach (for instance, tuning FV construction) to the various search task categories and seamless integration with the traditionally simple Web search interface.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented an approach that utilizes ontologies to enhance the effectiveness of large-scale search systems for the Web. We have described how such systems can be enriched with adapted ontologies by computing a feature vector for each of the ontology entities that typically includes terms (words) that are associated with the entities. We have briefly described how these FVs are automatically constructed by utilizing the knowledge represented in the ontologies. Finally, we have evaluated the approach.

A prototype was developed and real users evaluated its performance. We used parts of the EvOQS framework [23] to assess ontology fitness and capability to improve ontology-based search. In the framework, evaluation criteria are connected to scenarios of use with a purpose to enhance particular search tasks. We have discussed results of the experiment showing how different ontology quality aspects improve ontology-driven Web search performance. We have found difference of 10.5% in improved performance due to enhanced quality of ontology.

As a future work we will research alternative methods for post-processing of the retrieved documents utilizing the semantic relations in the ontology for better ranking and navigation.

7. ACKNOWLEDGMENTS

This research work is partially funded by the Integrated Information Platform for reservoir and subsea production systems (IIP) project, which is supported by the Norwegian Research Council (NFR). NFR project number 163457/S30. In addition, we would like to thank Jon Atle Gulla (NTNU), Per Gunnar Auran (Yahoo!), and Robert Engels (ESIS) for their support and help.

8. REFERENCES

[1] Aula, A.: Query formulation in Web information search. In: IADIS International Conference WWW/Internet (ICWI 2003), Algarve, Portugal, 403-410. IADIS (2003)

[2] Bonino, D., Corno, F., Farinetti, L., Bosca, A.: Ontology driven semantic search. *WSEAS Transaction on Information Science and Application* 1(6), 1597-1605. (2004)

[3] Brasethvik, T.: Conceptual modelling for domain specific document description and retrieval: An approach to semantic document modelling. PhD thesis, NTNU, Trondheim, Norway (2004)

[4] Castells, P., Fernandez, M., Vallet, D.: An adaptation of the vector-space model for ontology-based information retrieval. *IEEE Transactions on Knowledge and Data Engineering* 19(2), 261-272. (2007)

[5] Chang, Y., Ounis, I., Kim, M.: Query reformulation using automatically generated query entities from a document space. *Information Processing and Management* 42 (2006) 453-468

[6] d'Aquin, M., Motta, E., Sabou, M., Angeletou, S., Gridinoc, L., Lopez, V., Guidi, D.: Toward a New Generation of Semantic Web Applications. IEEE Intelligent Systems 23 (2008) 20-28

[7] Esmaili, K.S., Abolhassani, H.: A Categorization Scheme for Semantic Web Search Engines. Computer Systems and Applications, 2006. IEEE International Conference on. (2006) 171-178

[8] Formica, A., Missikoff, M., Pourabbas, E., Taglino, F.: Weighted Ontology for Semantic Search. OTM 2008, Part II, LNCS 5332, Springer-Verlag, (2008) 1289-1303

[9] Guha, R., McCool, R., Miller, E.: Semantic search. In: Proceedings of WWW2003, ACM Press (2003) 700-709

[10] Gulla, J.A., Borch, H.O., and Ingvaldsen, J.E.: Ontology Learning for Search ApplicationsOTM 2007, Part I, LNCS 4803, Springer-Verlag (2007) 1050-1062

[11] Jansen, B., Spink, A., Saracevic, T.: Real life, real users, and real needs: A study and analysis of user queries on the Web. Information Processing and Management 36(2) (2000)207-227

[12] Jiang, X., Tan, A.-H.: OntoSearch: A Full-Text Search Engine for the Semantic Web. In proc. of the 21st National Conf. on Artificial Intelligence and the 18th Innovative Applications of Artificial Intelligence (2006)

[13] Mangold, C.: A survey and classification of semantic search approaches. *Int. J. Metadata, Semantics and Ontologies* 2 (2007) 23-34

[14] McGuinness, D.: Question answering on the Semantic Web. IEEE Intelligent Systems 19(1) (2004) 82-85

[15] Nagypal, G.: Improving information retrieval effectiveness by using domain knowledge stored in ontologies. In: OTM Workshops. LNCS 3762, Springer-Verlag (2005) 780-789.

[16] Ogden, C.K., Richards, I.A.: The meaning of meaning: a study of the influence of language upon thought and of the science of symbolism. Kegan Paul, Trench, Trubner & Co, London (1930)

[17] Panagis, Y., Sakkopoulos, E., Garofalakis, J. and Tsakalidis, A. Optimisation mechanism for Web search results using topic knowledge. *Int. J. Knowledge and Learning* 2(1/2), (2006) 140–153.

[18] Paralic, J., Kostial, I.: Ontology-based information retrieval. In: Proceedings of the 14th International Conference on Information and Intelligent systems (IIS 2003), Varazdin, Croatia, (2003) 23-28.

[19] Rose, D., Levinson, D.: Understanding user goals in Web search. In: Proceedings of WWW2004, ACM Press (2004) 13-19

[20] Scheir, P., Pammer, V., Lindstaedt, S.N.: Information Retrieval on the Semantic Web - Does it exist? : LWA. Martin-Luther-University Halle-Wittenberg (2007)

[21] Schumacher, K., Sintek, M., and Sauermann, L.: Combining Fact and Document Retrieval with Spreading Activation for Semantic Desktop Search, *ESWC 2008*, LNCS 5021, Springer-Verlag (2008) 569-583.

[22] Solskinnsbakk, G., Gulla, J.A.: Ontological Profiles in Enterprise Search. EKAW 2008, LNAI 5268, Springer-Verlag (2008) 302-317.

[23] Strasunskas, D., Tomassen, S.L. The role of ontology in enhancing semantic searches: the EvOQS framework and its initial validation. *Int. J. Knowledge and Learning*, *4* (4) (2008) 398-414.

[24] Tomassen, S.L., Strasunskas, D. Construction of Ontology based Semantic-Linguistic Feature Vectors for Searching: the Process and Effect. In *IEEE/WIC/ACM Int. Conf. on Web Int. and Int. Agent Technology (WI-IAT '09)*, IEEE Computer Society, Milano, Italy (2009) 133-138.

[25] Tomassen, S.L.: Searching with Document Space Adapted Ontologies. In: Lytras, M.D., Carroll, J.M., Damiani, E., Tennyson, R.D. (eds.): *WSKS 2008*, Athens, Greece, September 24-26, 2008. Proceedings, Vol. 5288. Springer-Verlag, Athens, Greece (2008) 513-522

[26] Wang, H., Zhang, K., Liu, Q., Tran, T., and Yu, Y.: Q2Semantic: A Lightweight Keyword Interface to Semantic Search, *ESWC 2008*, LNCS 5021, Springer-Verlag (2008) 584-598.

[27] Yang, H-C. A method for automatic construction of learning contents in semantic Web by a text mining approach. *Int. J. Knowledge and Learning* 2(1/2) (2006) 89–105

.