Domain Specific Data Retrieval on the Semantic Web

Tuukka Ruotsalo[1,2,3]

[1] School of Information, University of California, Berkeley, USA
[2] Department of Media Technology, Aalto University, Finland
[3] Helsinki Institute for Information Technology (HIIT), Finland
tuukka.ruotsalo@aalto.fi

Abstract. The Web content increasingly consists of structured domain specific data published in the Linked Open Data (LOD) cloud. Data collections in this cloud are by definition from different domains and indexed with domain specific ontologies and schemas. Such data requires retrieval methods that are effective for domain specific collections annotated with semantic structure. Unlike previous research, we introduce a retrieval framework based on the well known vector space model of information retrieval to fully support retrieval of Semantic Web data described in the Resource Description Framework (RDF) language. We propose an indexing structure, a ranking method, and a way to incorporate reasoning and query expansion in the framework. We evaluate the approach in ad-hoc retrieval using two domain specific data collections. Compared to a baseline, where no reasoning or query expansion is used, experimental results show up to 76% improvement when an optimal combination of reasoning and query expansion is used.

1 Introduction

Search engines have revolutionized the way we search and fetch information by being able to automatically locate documents on the Web. Search engines are mostly used to locate text documents that match queries expressed as a set of keywords. Recently, the document centric Web has been complemented with structured metadata, such as the Linked Open Data cloud (LOD) [3]. In such datasets structured and semantic data descriptions complement the current Internet infrastructure through the use of machine understandable information provided as annotations [2]. Annotations are produced manually in many organizations, but automatic annotation has also become mature enough to work on Web scale [13]. As a result, we are witnessing increasing amount of structured data published on the Web.

Standards such as the RDF(S) [5] and publishing practices for linked data have enabled seamless access to structured Web data, but the underlying collections remain indexed using domain specific ontologies and schemas. In fact, such domain specific structure is the underlying element empowering the Semantic Web. For example, the data from cultural heritage data providers is very different from the data by scientific literature publishers, indexed with different vocabularies, and in the end, serving different information needs. As a result, different data collections are being published as a linked open data and accessed on the Web, but each individual publisher can decide about the semantics used to annotate the particular data collection. This imposes specific challenges for retrieval methods operating on such dataspace:

E. Simperl et al. (Eds.): ESWC 2012, LNCS 7295, pp. 422–436, 2012.

1. *Structured object data.* Search is targeted to objects or entities that are increasingly described using a combination of structured information and free text descriptions. For example, a tourist attraction could be described with information about the location of the site as coordinate data, the categorization of the site through references to a thesauri or an ontology, and the description of the attraction in free text format.
2. *Recall orientation.* A subset of the linked data cloud identified relevant for a specific application is often limited in size. Data collections are in hundreds of thousands or millions as opposite to billions as in conventional Web search. This favors recall oriented retrieval methods.
3. *Semantic gap between search and indexing vocabulary.* Objects originate form domain specific curated collections and are described using expert vocabulary. For example, a user searching for scientific objects inside a museum could be interested on spheres, galvanometers, and optical instruments, but could use terms ”science” and ”object” to express her information need.

To address the former challenges, we propose an extension of the Vector Space Model (VSM) [15] adapted to the RDF data model. Unlike in previous approaches [8,6,16,10,11,7], in our extension the indexing is based on RDF triples instead of individual concepts detected from text documents. The novelty of our model is that we use RDF triples as the basis for our indexing and ranking models instead of using ontologies only to expand individual terms in queries or text document indexing. This has only been addressed in [4], where horizontal indices were used to index RDF data. While similar in nature, in addition, we compare different query types, query complexity levels, and query expansion levels for the triple-based model. We also consider more complex queries than keyword queries as used in the previous work. In our approach, the queries can be any combination of keywords, triples or resources. In addition, the effect of query and document expansion, that allow background knowledge to be used as basis to reduce data sparsity and enable semantic indexing, are key contributions in our study.

We evaluate our approach in domain specific data retrieval on cultural heritage data collections and show that our adaptation of the VSM, combined with reasoning and correct query expansion strategy, yields to superior retrieval performance with an increase of 76% in mean average precision compared to a baseline approach. The rest of the paper is structured as follows. In section 2 we present the retrieval framework including indexing, retrieval and query expansion methods. In section 3 we explain the experimental setup, data and evaluation measures. Section 4 presents the results. Finally, we conclude with discussion, related work and future research directions.

2 Retrieval Framework

We use a retrieval framework based on the VSM and extend it to utilize RDF triples as indexing features. We show how indexing can be done for RDF triples, cosine similarity computed over such data representation, and how reasoning and query expansion can be incorporated in to the retrieval framework.

2.1 Data Representation

We start with a retrieval method based on the well known Vector Space Model of information retrieval [15]. We use metadata expressed as ontology-based annotations and utilize RDF as a representation language. RDF describes data as triples, where each triple value can be either a resource R or literal L. The feasibility of the index can be problematic in terms of the size of the triple-space if the triples would be directly used as indexing features. Using the pure VSM of information retrieval would cause the dimensionality of the document representation to be vectors that have occurrences for every deduced triple. The maximum dimensionality being the number of possible triples on the domain $T \in R \times R \times (R \cup L)$. It is well known that high dimensionality often causes problems in similarity measurements and has been recognized to be problematic in ontology-based search [6,1]. This would hurt the performance of the VSM, because many of the matching concepts would be the same in the tail of super concepts, i.e. almost all documents would be indexed using the triples consisting of resources appearing in the upper levels of the ontology hierarchies.

We reformulate the indexing of the documents and the triples in the deductive closure of their annotations as vectors describing occurrences of each triple given the property of the triple. Splitting the vector space based on property is not a new idea, but has been recently used in RDF indexing [12,4]. An intuition behind this is that properties often specify the point of view to the entity. For example, annotating *Europe* as a manufacturing place or subject matter should lead to completely different weighting of the resource, depending on the commonality of *Europe* as a subject matter or as a manufacturing place. In addition, properties are not expected to be used as query terms alone, but only combined with either subjects or objects of the triple. For example, it is unlikely that a user would express her information need by inserting a query to return all documents with *dc:subject* in the annotations. However, a user could construct a query that would request all documents with *dc:subject* having a value *Europe*. Literals are treated separately from concepts. We tokenize literals to words and stem them using the Porter stemming method. After this they are stored in the same vectors as the concepts. In practice, the data is often described using a schema, where the subject of the triple is the identifier for the entity being described, as in the data used in our experiments. However, our indexing strategy enables indexing of arbitrary RDF graphs.

Accessing the correct index for each vector space fast in the query phase requires an external index. For this purpose we define a posting list that maps the index of the correct vector space to the query. We propose a model over possible vector spaces, first one for every possible property, and two additional vector spaces for subject and object. From now on we refer to these actually indexed subjects and objects as concepts to avoid mixing these with the subject of an RDF triple. Every concept is indexed in a vector space that defines the occurrence of the concept in an annotation of a specific entity. These vector spaces are referred as y and they form a set of vector spaces Y with a length x, i.e. $Y_x = \{y_1, ..., y_x\}$.

This indexing strategy requires a large number of vector spaces, but the triple dimension of each matrix is lower because the maximum term length k for triples is the number of resources and literals R, and for the document dimension only the documents that have triples in the particular vector space are indexed. This avoids the high dimensionality problem when computing similarity estimates.

2.2 Weighting

The purpose of the indexing strategy is not only to reduce the dimensionality to make computation faster, but also to enable more accurate weighting by avoiding the problems caused by the high dimensionality. Intuitively, some of the triples are likely to be much less relevant for the ranking than others. For example, matching a query only based on a triple *<rdf:Resource, rdf:Resource, rdf:Resource>* will lead to a match to all documents, but is meaningless for search purposes. On the other hand, a resource Helsinki, should be matched to all documents indexed with resource Helsinki, but also to the documents indexed with Europe, because they belong into the same deductive closure, but with smaller weight. For this purpose we use *tf-idf* weighting over the resources within a specific vector space. In normalized form *tf* is:

$$tf_{i,j} = (\frac{N_{i,j}}{\sum_k N_{k,j}})^{\frac{1}{2}}, \tag{1}$$

where $N_{i,j}$ is the number of times a resource i is mentioned in the vector space of document j and $\sum_k N_{k,j}$ is the sum of the number of occurrences of all resources of the document j. In a similar way, inverse document frequency *idf* is defined as:

$$idf_i = 1 + log(\frac{N}{n_i + 1}), \tag{2}$$

where n_i is the number of documents, where the resource i appears within the specific vector space and N is the total number of documents in the system. The weight of an individual resource in a specific vector space is given by:

$$w_{i,j} = tf_{i,j} \cdot idf_i. \tag{3}$$

The *tf-idf* effect in triple-space is achieved based on the annotation mass on resources, but also through reasoning. For example, the resource Europe is likely to have much more occurrences in the index than the resource Finland, since the index contains the deductive closures of the triples from annotations using resource identifiers also for other European countries. This makes the *idf* value for resource Finland higher than for resource Europe. A document annotated with resources Germany, France and Finland would increase the *tf* value for the resource Europe, because through deductive reasoning Germany, France and Finland are a part of Europe. Naturally, the *tf* could also be higher in case the document is annotated with several occurrences of the same resource, for example as a result of automatic annotation procedure based on text analysis [13].

2.3 Ranking

In the vector model the triple vectors can be used to compute the degree of similarity between each document d stored in the system and the query q. The vector model evaluates the similarity between the vector representing an individual document V_{d_j} and a query V_q. We reformulate the cosine similarity to take into account a set of vector spaces, one for each possible combination of triples given the models $y \in Y$ as opposite to the classic VSM that would use only one vector space for all features. For this purpose, we adopt the modified cosine similarity ranking formula used in Apache Lucene open source search engine[1], where the normalization based on Euclidean distance is replaced with a length norm and a coord-factor. The length norm is computed as:

$$ln(V_{d_j,y}) = \frac{1}{\sqrt{nf}}, \tag{4}$$

where nf is the number of features present used to index the document d_j in the vector space y under interest. The coord-factor is computed as:

$$cf(q, d_j) = \frac{mf}{k}, \tag{5}$$

where mf is the number of matching features in all vector spaces for document d_j and query q and k is the total number of features in the query.

In our use case, these have two clear advantages compared to the classic cosine similarity. First, the use of the length norm gives more value to documents with less triple occurrences within the vector space under interest. In our case this means that documents annotated with less triples within a particular vector space y get relatively higher similarity score. This is intuitive, because the knowledge-base could contain manually annotated documents with only few triples and automatically annotated documents with dozens of triples. In addition, some vector spaces can end up having more triples, as a result of reasoning or more intense annotation, than others. The number of matching features in queries also should increase the similarity of the query and document. This effect is captured by the coord-factor. We can now write the similarity as:

$$sim(q, d_j) = cf(q, d_j) \cdot \sum_{y=1}^{x} \sum_{i=1}^{k} (w_{i,y_j} \cdot ln(V_{y_{d_j}})), \tag{6}$$

where the dot product of the vectors now determines the weight $w_{i,j}$ and is computed across all vector spaces y. In this way the ranking formula enables several vector spaces to represent a single document because length norm is computed for each vector space separately. This can be directly used to operate with our triple space indexing.

The model approximates the importance of all the different combinations of $y \in Y$ separately. Intuitively, this is a coherent approach: the importance of a concept in the domain is dependent on the use of the concept in a triple context. Note that our approach does not normalize across the vector spaces. This favors matches in several vector space instead of a number of matches in a single vector space. For example, a query with several triples with the property *dc:subject* and a single triple with the property *dc:creator* would favor queries that have both *dc:subject* and *dc:creator* present over queries that would have matches only for one of the properties.

[1] The features of the similarity computation that are not used in our method and experiments are omitted. The full description of the original ranking formula can be found at http://lucene.apache.org/

2.4 Reasoning and Query Expansion

The adaptation of the vector space model that we presented in the earlier section assumes the existence of document vectors that can be then stored in separate vector spaces. RDF(S) semantics enable deductive reasoning on the triple space. Using such information in the indexing phase is often called document expansion. This means that the document vectors are constructed based on the triple-space resulting from a deductive reasoning process.

For example, an annotation with an object Paris, could be predicated by different properties. One document could be created in Paris while another document could have Paris as a subject matter. Through deductive reasoning both of these annotation triples are deduced to a triple, where the property pointing to the concept Paris is *rdf:Resource*. In a similar way, the concept Paris can be deduced through subsumption reasoning to France, Europe, and so on.

If a search engine receives a query about Paris, it should not matter for the search engine whether the user is interested in Paris in the role of subject matter or place of creation. Therefore, the search engine should rank these cases equally based on only the information that the documents are somehow related to Paris. In other words, based on the triple, where the property is *rdf:Resource*. On the other hand, if the user specifies an interest in Paris as a subject matter, the documents annotated in such way can be ranked higher by matching them to a vector space of subject matters. This functionality is already enabled using the vector space model for triple space by indexing deductive closures along with the original triples.

Another way to improve the accuracy of the method is ontology-based query expansion. While deductive reasoning provides logical deduction based on the relations available in the ontologies, the user can be interested also in other related documents. For example, users interested in landscape paintings, could also be interested on seascape paintings, landscape photographs and so on. These can be related in the ontology further away or with different relationships that are included in the standard RDF(S) reasoning.

Ontologies can be very unbalanced and depending on the concepts used in the annotation, different level of query expansion may be necessary. For example, a document annotated with a concept Buildings may already be general enough and matches to many types of buildings, while a document annotated with the concept Churches might indicate user's interest, not only on churches, but also other types of religious buildings.

Measuring a concept to be semantically close to another concept, and therefore a good candidate for the query expansion, can be approximated using its position in the ontological hierarchy [14]. The more specific the concept is, more expansion can be allowed. We use the Wu-Palmer measure to measure the importance of a resource (subject, predicate, and object separately) given the original resource in the query triple. Formally, the Wu-Palmer measure for resources c and c' is:

$$rel_{WP}(c,c') = \frac{2l(s(c,c'),r)}{l(c,s(c,c')) + l(c',s(c,c')) + 2l(s(c,c'),r)}, \tag{7}$$

where $l(c,c')$ is a function that returns the smallest number of nodes on the path connecting c and c' (including c and c' themselves), $s(c,c')$ is a function that returns the lowest common super-resource of resources c and c', and r is the root resource of the ontology.

The resources having a Wu-Palmer value above a certain threshold are selected for query expansion. We construct all the possible triples that are possible based on the resources determined by the Wu-Palmer measure and select the most general triples as the expanded triples that are used in the actual similarity computation. This means that all subjects, properties, and objects of any triple in the query are included by using all permutations of the resources in these resulting sets and the most general combination is selected. By the most general combination, we mean triple that has the longest distance in terms of subsumption from the original triple in terms of the expanded subject, predicate and object, each measured individually. This also removes possible redundancy of the original query triples, such as inclusion of triples.

In case other relations are used in the expansion, all of the triples are included. In other words, we include only the most general case in terms of subsumption, but include related terms as new triples. The rationale behind including only the most general triple is that including all possible super-triples could lead to a substantial amount of matching triples and may hurt the accuracy of the similarity computation.

The Wu-Palmer measure can be used to dynamically control the query expansion level towards an index of concepts that form a tree. Such a tree can be constructed in many different ways. A trivial case is to use only subsumption hierarchies, a semantically coherent taxonomy of concepts. However, ontologies enable also other relations to be used in query expansion. We refer different combinations of such relations as the query expansion strategy.

We investigate the following query expansion strategies: related terms only, subsumption only, full expansion. Related terms only strategy means that a semantic clique is formed based on the nodes directly related to the concept being expanded (distance of arcs is one), but no subsumption reasoning is used. Subsumption only strategy means that the query is expanded using transitive reasoning in subsumption hierarchies. This means that additional query expansion to other concepts than those in the deductive closure can be done only using subsumption hierarchies. Full expansion means that both, subsumption and related terms are used in expansion and the tree index is built using subsumption relations and related terms of each concept achieved through subsumption. Related terms are not treated as transitive.

For example, using only the subsumption hierarchies, we could deduce the information that the concept ”landscape paintings” is related to its superconcept ”paintings” and through that to the concept ”seascape paintings”, because they have a common superconcept. Using the full expansion we could obtain an additional information that ”seascape paintings” is further related to ”seascapes”, ”marinas” and so on.

3 Experiments

We conducted a set of laboratory experiments to determine the retrieval performance of the method and the effect of different indexing and query expansion strategies.

3.1 Method Variants

We created different variants of the VSM method by varying indexing strategies and query expansion levels. These were used to study the effect of the different combinations to the retrieval performance. The performance of all different indexing strategies was measured separately: related terms only, subsumption only, and full expansion. All of the strategies were then measured using different levels of query expansion by varying the Wu-Palmer measure from 1.0 to 0.1. All of the strategies were implemented on top of the triple-space index.

3.2 Data

We used a dataset in the domain of cultural heritage, where the documents have high quality annotations. The dataset consists of documents that describe museum items, including artwork, fine arts and scientific instruments, and points of interest, such as visiting locations, statues, and museums. The data was obtained from the Museo Galileo in Florence, Italy, and from the Heritage Malta. The document annotations utilize the Dublin Core properties and required extensions for the cultural heritage domain, such as material, object type, and place of creation of the document described. An example annotation of a document describing a scientific instrument from the Museo Galileo is described in Figure 1.

```
<dc:identifier> <urn:imss:instrument:402015> .
<sm:physicalLocation> <http://www.imss.fi.it/> .
<dc:title> "Horizontal dial" .
<dc:subject> "Measuring time" .
<dc:description> "Sundial, complete with gnomon..." .
<dc:subject> <aat:300054534> . (Astronomy)
<sm:dateOfCreation> <sm:time_1501_1600> . (16th Century)
<sm:material> <aat:300010946> . (Gilt Brass)
<sm:objectType> <aat:300041614> . (Sundial)
<sm:placeOfCreation>  <tgn:7000084> (Germany)
<sm:processesAndTechniques> <aat:300053789> . (Gilding)
<dc:terms/isPartOf> "Medici collections" .
<rdf:type> <sm:Instrument> .
```

Fig. 1. An example of the data used in the experiments. Subjects of the triples are all identifiers of the resource being describes and are therefore omitted. Description is shortened.

The documents are indexed with RDF(S) versions of Getty Vocabularies[2]. The RDF(S) versions of the Getty Vocabularies are lightweight ontologies that are transformed to RDF(S) from the original vocabularies, where concepts are organized in subsumption hierarchies and have related term relations. Geographical instances are structured in meronymical hierarchies that represent geographical inclusion. Temporal data is described using a hand crafted ontology that has concepts for each year, decade,

[2] http://www.getty.edu/research/conducting_research/vocabularies/

century, and millennium organized in a hierarchy. Literal values are indexed in the VSM as Porter stemmed tokenized words.

3.3 Queries and Relevance Assessments

The query set consists of 40 queries that were defined by domain experts in the same museums where the datasets were curated. Figure 2 shows two example queries, one for astronomers and subject matter optics, and another for physicist Leopoldo Nobili and subject matter of galvanometers, batteries and electrical engineering. Relevance assessments corresponding to the query set were provided for a set of 500 documents in both museums. Museum professionals provided relevance assessments for the dataset by assessing each document either relevant or not relevant separately for all of the queries. The dataset and relevance assessment were carried out specifically for this study. This is a relatively large set of queries and relevance assessments for one-off experiment because the recall is analyzed with full coverage by domain experts meaning that all of the documents are manually inspected against all of the queries. Pooling or automatic pre-filtering was not used. This makes the relevance assessments highly reliable, avoids bias caused by automatic pre-filtering, and takes into account all possible semantic relevance, even non-trivial connections judged relevant by the domain experts.

The domain experts were asked to created queries typical for the domain, such that the queries would include also non-trivial queries considering the underlying collection. For example, a query containing the concept ”seascapes” was judged relevant also for objects annotated with the concept ”landscape paintings”, and for objects annotated with ”marinas”, ”boats”, ”harbors” and so on. The judges were allowed to inspect the textual description in addition to the image of the objects when assessing relevance.

```
<rdf:Resource> <aat:300025789> . (astronomers)
<dc:subject> <aat:300134506> . (astronomical photography)
<dc:subject> <aat:300211119> . (optical toys)
<dc:subject> <aat:300056210> . (optical properties)
<rdf:type> <sm:Instrument> .


<rdf:Resource> "Leopoldo Nobili" .
<dc:subject> <aat:300197519> . (galvanometers)
<dc:subject> <aat:300002501> . (batteries)
<dc:subject> <aat:300054490> . (electrical engineering)
<rdf:type> <sm:Instrument>
```

Fig. 2. An example of two sets of queries defined by experts in the Museo Galileo. The namespace *dc* and *sm* refer to the Dublin Core and a custom extension of the Dublin Core properties for the cultural heritage domain, and *aat* to the Art and Architecture Thesaurus of the Getty Foundation. The subject of each RDF triple is omitted, because it is rdf:Resource for these queries.

3.4 Evaluation Metrics

The accuracy of the retrieval methods was measured using Recall, Precision and Mean Average Precision. In addition, we plotted non-interpolated precision-recall curves on 11 recall levels to get an understanding of the performance differences between the methods. Recall R is defined as the number of relevant documents retrieved by a search method divided by the total number of relevant documents in the system, while precision P is defined as the number of relevant documents retrieved by a search method divided by the total number of documents retrieved.

Precision and recall are vulnerable measures because often when precision increases, recall decreases and vice versa. Therefore, a single measure that can be used to estimate a balanced performance in terms of precision and recall can be useful. We are interested also on ranking and a natural measure to be used to investigate the ranking along with the precision and recall tradeoff is Mean Average Precision (MAP). MAP for a set of relevant documents $\{d_1, ..., d_{m_j}\}$ for a query q_j of total Q queries and a set of ranked retrieval results RA_{jk} from the top result until one gets to document d_k is:

$$MAP(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k}^{m_j} P(RA_{jk}). \tag{8}$$

When a relevant document is not retrieved at all, the precision value in the above equation is taken to be 0. For a single information need, the average precision is the average area under the precision-recall curve for a set of queries.

3.5 Statistical Significance

The statistical significance of the differences of the results obtained using different combinations of methods were ensured using the the Friedman test. The Friedman test is a non-parametric test based on ranks and is suitable for comparing more than two related samples. The statistical significance between method pairs was then analyzed using a paired Wilcoxon Signed-Rank test with Bonferonni correction as a post-hoc test. The differences between the method variants were found to be statistically significant (p<0.001). The Friedman test was chosen because the data was not found to be normally distributed using the Shapiro and Wilk test.

4 Results

Figures 3 and 4 summarize the results. Figure 3 presents the precision - recall using each method variant when no query expansion was used. The curve on Figure 4 presents the same results for the best query expansion determined by the Wu-Palmer cutoff that was found to lead to best MAP for each method variant. In other words, the best achieved indexing strategy - query expansion combination. The following main findings can be observed. First, using the full indexing leads to the best overall performance. It performs equally good to subsumption indexing when a combination of query expansion and reasoning is used, but outperforms the subsumption indexing on low recall levels and

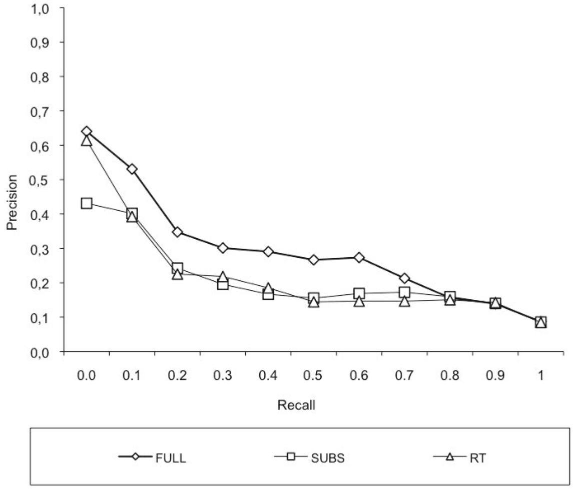

Fig. 3. Precision plotted on 11 recall levels for different reasoning and indexing strategies. No query expansion is used. The values are averaged over the 40 queries.

in the case where no query expansion is used. Second, the subsumption indexing and full indexing outperform related terms indexing in all tasks. The results show up to 76% improvement compared to a variation where no reasoning and query expansion are used.

Both, indexing using subsumption and full indexing, that also uses subsumption, seems to perform clearly better than related term indexing. The performance is increased by 0.15 (68%) in MAP compared to the baseline. The gain in performance achieved using subsumption and full indexing strategies imply that subsumption reasoning is the most important factor affecting the accuracy of the retrieval. Full indexing strategy clearly outperforms the other strategies on overall performance and performs best even on the lowest recall levels. An interesting finding is that subsumption reasoning and indexing strategy do perform worse than related terms strategy on the low recall levels, when reasoning is not complemented with query expansion.

Query expansion has a significant overall effect and, in addition to reasoning, is an important factor affecting the accuracy of the retrieval. Query expansion increases the accuracy up to 0.16 (76%) in terms of MAP when full expansion reasoning and indexing strategy is used. It is notable that the subsumption reasoning and indexing strategy actually performs only equally good compared to the baseline approach when no additional query expansion is used. This indicates that the combination of correct indexing strategy and query expansion is crucial to achieve optimal accuracy. An additional query expansion using super concepts from ontologies was found to be most effective when using cut-off value 0.9 to 0.7 of the Wu-Palmer measure. This means an expansion of zero to three nodes in the ontology graph in addition to the standard RDF(S) reasoning.

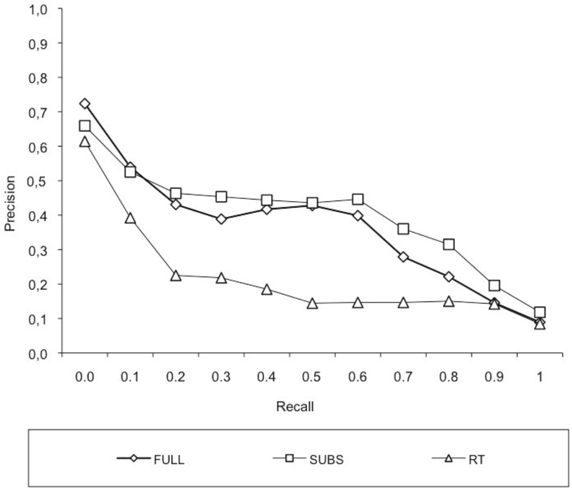

Fig. 4. Precision plotted on 11 recall levels for different reasoning and indexing strategies. The best combination of query expansion and reasoning is used. The values are averaged over the 40 queries.

It is observable in the results that the gold standard and queries favor recall-oriented methods, which was expected to be the case in domain specific setting. For example, the subsumption indexing strategy with the Wu-Palmer cut-off at 0.4 leads to an equally good performance as the cut-off 0.7, while cut-off values 0.5 and 0.6 perform worse. This indicates that using extensive query expansion compensates better semantic approximation achieved using related term relations together with subsumption reasoning. We believe that this is due to the fact that our data set consists of documents from a relatively specific domain and a collection of only 1000 documents. In additional runs we observed that precision - recall curves have different tradeoffs when varying the Wu-Palmer cut-off values. Using more query expansion increases recall, but does not hurt precision as extensively as could be expected. Our conclusion is that our dataset favors recall oriented approaches without a serious precision trade-off. This may not be the case in settings, where data is retrieved from a data cloud that is linked to other domains. Therefore, we believe that full expansion with mild query expansion leads to best overall performance.

5 Conclusions and Discussion

In this paper, we propose an indexing and retrieval framework for structured Web data to support domain specific retrieval of RDF data. The framework is computationally feasible because it avoids the high dimensionality of the triple space in similarity computation by using triple based indexing. We conducted a set of experiments to validate the performance of the approach and combine different reasoning, indexing and query expansion strategies. We show that ontology-based query expansion and reasoning improves retrieval in Semantic Web data retrieval and can be effectively used in our adaptation of the vector space model. We also provide empirical evidence to support the effect of self-tuning query expansion method that is based on a metric that measures the depth of the ontology graphs. The experimental evaluation of the framework led to the following conclusions:

1. The best combination of reasoning and query expansion leads to improvement of accuracy up to 76%.
2. Full reasoning and indexing strategy improves accuracy of retrieval, with best results achieved when combined with query-expansion.
3. Query expansion that considers also other relations than those belonging to the standard RDF(S) reasoning improves results. The Wu-Palmer cut-off values around 0.7-0.9 when combined with full indexing leads to best results.
4. Using only subsumption indexing seems to work relatively well in our experiments, but requires extensive expansion. This can be problematic with more diverse datasets than the ones used in this study.

We conducted experiments that tested a number of different techniques and their combinations. However, the experimental setup leaves room for further research. While we used two separate collections and queries from different annotators and institutions, these were indexed using the same ontologies. The data used in the experiments is from the cultural heritage domain and may not generalize to other more open domains.

We measured the performance of the methods against expert created gold standard on a set of domain specific annotations on the cultural heritage domain. The relevance assessments are determined manually for the whole dataset, unlike in some other datasets proposed for semantic search evaluation, such as the Semantic Search Workshop data [9], where the relevance assessments were determined by assessing relevance for documents pooled form 100 top results from each of the participating systems, queries were very short, and in text format. This ensures that our dataset enables measuring recall and all of the query-document matches, even non-trivial, are present. The set of queries, for which the relevance assessments were created, are in the form of sets of triples. This avoids the problems of query construction and disambiguation of the terms, which means that we are able to measure the retrieval performance independently of the user interface or initial query construction method. While we realize that disambiguation and query construction are essential for search engines, we think that they are problems of their own to be tackled by the Semantic Search community. Our methods are therefore valid for information filtering scenarios and search scenarios that can use novel query construction methods, such as faceted search, or query suggestion techniques. The methods only operate on numerical space for triples and implements ranking independently from the specific RDF dataset it could also be implemented as a ranking layer under database management systems that support more formal query languages such as SPARQL. Our experiments were run on a gold standard acquired specifically for this study, that makes the results more reliable and the gold standard highly reliable. However, we used relatively small dataset of 1000 documents which makes the task recall oriented. However, the methods themselves scale to large collections, because the indexing and retrieval framework does not make any assumptions over the classic VSM and are able to delimit the dimensionality of the VSM based on splitting the space separately for each property. However, small collections are typical in domain specific search and the results may not be directly comparable with results obtained for other collections. While full query expansion with subsumption reasoning works well for such a homogenous dataset, this might not be true for more varying datasets. This is due to the fact that the best performance was achieved with the Wu-Palmer cut-off value of 0.4 that allows traversing the supertree of a concept for several nodes. This could hurt the accuracy when applied to larger precision oriented datasets. However, our results are a clear indication of the effectiveness of both query expansion and reasoning.

Furthermore, ontologies are not the only source for semantic information. Our method operates in pure numerical vector space that makes it possible to apply standard dimensionality reduction and topic modeling methods that could reveal the semantics based on collection statistics. Since our experiments showed that maximal query expansion using ontologies leads to best retrieval accuracy, such methods are an interesting future research direction.

Acknowledgements. The work was conducted under the Academy of Finland Grant (135536), Fulbright Technology Industries of Finland Grant, and Finnish Foundation for Technology Promotion Grant.

References

1. Agirre, E., Arregi, X., Otegi, A.: Document expansion based on wordnet for robust ir. In: Proceedings of the 23rd International Conference on Computational Linguistics: Posters, COLING 2010, pp. 9–17. Association for Computational Linguistics, Stroudsburg (2010)
2. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web: Scientific American. Scientific American (May 2001)
3. Bizer, C., Heath, T., Berners-Lee, T.: Linked data - the story so far. Int. J. Semantic Web Inf. Syst. 5(3), 1–22 (2009)
4. Blanco, R., Mika, P., Vigna, S.: Effective and Efficient Entity Search in RDF Data. In: Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A., Kagal, L., Noy, N., Blomqvist, E. (eds.) ISWC 2011, Part I. LNCS, vol. 7031, pp. 83–97. Springer, Heidelberg (2011)
5. Brickley, D., Guha, R.V.: RDF vocabulary description language 1.0: RDF Schema W3C recommendation. Recommendation, World Wide Web Consortium (February 10, 2004)
6. 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)
7. Fazzinga, B., Gianforme, G., Gottlob, G., Lukasiewicz, T.: Semantic web search based on ontological conjunctive queries. Web Semantics: Science, Services and Agents on the World Wide Web 9(4), 453–473 (2011)
8. Férnandez, M., Cantador, I., López, V., Vallet, D., Castells, P., Motta, E.: Semantically enhanced information retrieval: An ontology-based approach. Web Semantics: Science, Services and Agents on the World Wide Web 9(4), 434–452 (2011)

9. Halpin, H., Herzig, D., Mika, P., Blanco, R., Pound, J., Thompon, H., Duc, T.T.: Evaluating ad-hoc object retrieval. In: Proceedings of the International Workshop on Evaluation of Semantic Technologies, Shanghai, China. CEUR, vol. 666 (November 2010)
10. Kiryakov, A., Popov, B., Ognyanoff, D., Manov, D., Kirilov, A., Goranov, M.: Semantic Annotation, Indexing, and Retrieval. In: Fensel, D., Sycara, K., Mylopoulos, J. (eds.) ISWC 2003. LNCS, vol. 2870, pp. 484–499. Springer, Heidelberg (2003)
11. Ning, X., Jin, H., Jia, W., Yuan, P.: Practical and effective ir-style keyword search over semantic web. Information Processing & Management 45(2), 263–271 (2009)
12. Pérez-Agüera, J.R., Arroyo, J., Greenberg, J., Iglesias, J.P., Fresno, V.: Using bm25f for semantic search. In: Proceedings of the 3rd International Semantic Search Workshop, SEMSEARCH 2010, pp. 2:1–2:8. ACM, New York (2010)
13. Ruotsalo, T., Aroyo, L., Schreiber, G.: Knowledge-based linguistic annotation of digital cultural heritage collections. IEEE Intelligent Systems 24(2), 64–75 (2009)
14. Ruotsalo, T., Mäkelä, E.: A comparison of corpus-based and structural methods on approximation of semantic relatedness in ontologies. International Journal on Semantic Web and Information Systems 5(4), 39–56 (2009)
15. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Communications of the ACM 18(11), 613–620 (1975)
16. Vallet, D., Fernández, M., Castells, P.: An Ontology-Based Information Retrieval Model. In: Gómez-Pérez, A., Euzenat, J. (eds.) ESWC 2005. LNCS, vol. 3532, pp. 455–470. Springer, Heidelberg (2005)