# Entity Search Evaluation over Structured Web Data

Roi Blanco
Yahoo! Research
Diagonal 177
Barcelona, Spain
roi@yahoo-inc.com

Harry Halpin
University of Edinburgh
10 Crichton St.
Edinburgh, UK
H.Halpin@ed.ac.uk

Daniel M. Herzig
Institute AIFB
Karlsruhe Institute of Technology
76128 Karlsruhe, Germany
herzig@kit.edu

Peter Mika
Yahoo! Research
Diagonal 177
Barcelona, Spain
pmika@yahoo-inc.com

Jeffrey Pound
David R. Cheriton School of Computer Science
University of Waterloo
Waterloo, Canada
jpound@cs.uwaterloo.ca

Henry S. Thompson
University of Edinburgh
10 Crichton St.
Edinburgh, UK
ht@inf.ed.ac.uk

## ABSTRACT

The search for entities is the most common search type on the web beside navigational searches. Whereas most common search techniques are based on the textual descriptions of web pages, semantic search approaches exploit the increasing amount of structured data on the Web in the form of annotations to web-pages and Linked Data. In many technologies, this structured data can consist of factual assertions about entities in which URIs are used to identify entities and their properties. The hypothesis is that this kind of structured data can improve entity search on the web. In order to test this hypothesis and to consistently progress in this field, a standardized evaluation is necessary. In this work, we discuss an evaluation campaign that specifically targets entity search over Linked Data by the means of keyword queries, including both queries that directly mention the entity as well as those that only describe the entities. We also discuss how crowd-sourcing was used to obtain relevance assessments from non-expert web users, the participating systems and the factors that contributed to positive results, and how the competition generalizes results from a previous crowd-sourced entity search evaluation.

## Categories and Subject Descriptors

H.3.3 [**Information Storage Systems**]: Information Retrieval Systems

## General Terms

Performance, Experimentation

## Keywords

entity search, search engines, retrieval, evaluation

## 1. INTRODUCTION

In any new information retrieval task, one of the most common components needed is regular and standardized evaluation in order to determine if progress is being made, and this is increasingly important as large-scale amounts of structured data about entities is being introduced to the Web. While common information retrieval search technique used by Web search engines rely primarily on information implicit in the textual descriptions of web-pages and the structure of the links between web-pages, an increasing amount of data is available on the Web using the RDF (Resource Description Format) standard, which attempts to make explicit information about entities and the relations between them in a structured form as to create what has been termed the "Semantic Web" by Tim Berners-Lee, the inventor of the World Wide Web [2]. A number of entity-centric search engines have independently arisen that search and index this data, but they have not been systematically evaluated using information retrieval metrics until the SemSearch competition in 2010 [9]. However, this competition was criticized as being too simplistic by focusing only on keyword queries for named entities. We focus here on the second round of the competition, which broadened the kinds of queries tested from simple keyword-based queries for entities to complex queries that contained criteria that matched multiple entities. The process of how crowd-sourced judges were used to determine entity-based relevance is detailed, and the systems that participated are described, with a focus on the factors that led to their success in searching for entities.


EOS, SIGIR 2011 workshop, July 28, Beijing, China

## 2. RDF AND THE SEMANTIC WEB

Semantic Web languages based on RDF identify entities by assigning URIs (Uniform Resource Identifiers), such as *http://www.example.org/Paris*, to both entities and their relationships to other entities. So, one could describe the fact that 'Paris is the capital of France' in RDF by stating the triple *(http://example.org/Paris , http://example.org/capital, http://example.org/France)*. The first URI is the *subject*, the second URI is the *predicate*, and the third (either another URI or a string with a data-value) is *object*. RDF is then a graph model where the vertices may be either URIs that name entities or text, and the edges between vertices are also labelled with URIs. When this RDF data is accessible over HTTP at these URIs, and so accessible to search engines, this RDF data is called "Linked Data"[1]. Overall, the amount of Linked Data accessible to search engines has

[1] `http://www.linkeddata.org`

grown to 10s of billions RDF statements. Further technologies, such as reasoning about RDF entities and classes using schemas, are also available. The hypothesis of the Semantic Web is that by creating a clearly defined format for sharing structured data across the Web a number of common tasks can be improved for the benefit of end users.

## 2.1 Entity Search on the Semantic Web

The search for entities is the most common search type on the Web after navigational searches [15]. However, searching entities in the predominant textual content of the web is hard, because it requires error-prone and expensive text processing techniques such as information extraction, entity identification, entity disambiguation over large collections of documents. One of the main goals of the Semantic Web is to make information available in a structured form that is easier and more reliable for machines to process than documents. The hypothesis is that this kind of structured data can improve entity search on the Web. In order to test this hypothesis and to consistently progress in this field, a standardized evaluation is necessary.

# 3. OVERVIEW OF THE CHALLENGE

As noted earlier, while there is an increasing amount of data about entities on the Web encoded in RDF and accessible as Linked Data, and a growing number of independent 'Semantic search' engines that specialize in crawling and searching RDF such as Sindice [13]. Assessing the relevance of the results provided by these *semantic search engines* require an information retrieval evaluation methodology over realistic data-sets. The first large-scale evaluation of these Semantic Search engines took place in 2010 [9], focusing on the task of entity search. This choice was driven by the observation that over 40% of queries in real query logs fall into this category [15], largely because users have learned that search engine relevance decreases with longer queries and have grown accustomed to reducing their query (at least initially) to the name of an entity. However, the major feedback and criticism of the 2010 SemSearch Challenge was that by limiting the evaluation to keyword search for named entities the evaluation excluded more complex searches that would hypothetically be enabled by semantic search over RDF. Therefore, the 2011 SemSearch competition introduced a second track, the "List Search" track, that focussed on queries where one or more entities could fulfill the criteria given to a search engine.

The Semantic Search challenge differs from other evaluation campaigns on entity search. In comparison to the TREC 2010 Entity Track [1], the SemSearch Challenge searches over structured dat a in RDF rather than text in unstructured web-pages and features more complex queries. Likewise, in comparison to the INEX Entity-Ranking task [7], SemSearch focusses on RDF as opposed to XML as a data-format, and searches for relevance over entire RDF descriptions, not passages extracted from XML. Unlike the QALD-1 Question Answering over Linked Data [16] task, our queries were not composed of hand-crafted natural language questions built around particular limited data-sets such as DBPedia and MusicBrainz (i.e. RDF exports of Wikipedia and music-related information), but of both simple and complex real-world queries from actual query logs. The use of queries from actual Web search logs is also a major difference between our competition and all aforementioned competitions such as TREC and INEX. Keyword search over structured data gets also more attention in the database community [10] and an evaluation framework was recently proposed [6], but an standardized evaluation campaign is not yet available. The Semantic Search Challenge comprised two different tracks, which are described in the next section.

## 3.1 Entity Search Track

The Entity Search Track aims to evaluate a typical search task on the web, keyword search where the keyword(s) is generally the name of the entity. Entities are ranked according to the degree to which they are relevant to the keyword query. This task has been the same as defined for the 2010 Semantic Search Challenge [9].

## 3.2 List Search Track

The List Search Track comprises queries that describe sets of entities, but where the relevant entities are not named explicitly in the query. This track was designed to encourage participating systems to exploit relations between entities and type information of entities, therefore raising the complexity of the queries. The information need is expressed by a number of keywords (minimum three) that describe criteria that need to be matched by the returned results. The goal is to rank higher the entities that match the criteria than entities that do not match the criteria. Examples of the queries used in the two tracks are shown in Table 1 and described in the next section.

# 4. EVALUATION METHODOLOGY

Evaluating different approaches against each other requires a controlled setting in order to achieve comparability as well as repeatability of the evaluation's results. For the evaluation of ranked results, the *Cranfield* methodology [5] is the de-facto standard in information retrieval. The Cranfield methodology measures retrieval effectiveness by the means of a fixed document collection, a set of queries, a set of relevance assessments denoting which documents are (not) relevant for a query, and a number of well-understood and defined metrics, such as precision and recall. As we are evaluating search over entities in the RDF data format, our evaluation setting requires as the result of each participating system a ranked list of entities from an RDF data collection in response to each queries. Thus, the units of retrieval are individual entities identified by URIs (Uniform Resource Identifier), not documents. In the following sections, we describe how we created a standardized setting consisting of a RDF data collection, a set of keyword queries garnered from real-world Web search logs, and crowd-sourced relevance assessments.

## 4.1 Data Collection

A standard evaluation data collection should be not biased towards any particular system or towards a specific domain, as our goal is to evaluate general purpose entity search over RDF data. Therefore, we needed a collection of documents that would be a realistically large approximation to the amount of RDF data available 'live' on the Web and that contained relevant information for the queries, while simultaneously of a size that could be manageable by the resources of a research groups. We chose the 'Billion Triples Challenge' 2009 data set, a data-set created for the

Semantic Web Challenge [3] in 2009. The dataset was created by crawling data from the web as well as combining the indexes from several semantic web search engines. The raw size of the data is 247GB uncompressed and it contains 1.4B RDF statements describing 114 million entities. The statements are composed of *quads*, where a quad is a four tuple comprising the four fields *subject*, *predicate*, *object*, as is standard in RDF, but also a URI for *context*, which basically extends a RDF triple with a new field giving a URI that the triples were retrieved from (i.e. hosted on). Details of the dataset can be found at `http://vmlion25.deri.ie/` and it is available for download at `http://km.aifb.kit.edu/ws/dataset_semsearch2010`. There was only a single a modification necessary for using this data-set for entity search evaluation which was to replace RDF blank nodes (an existential variable in RDF) with unique identifiers so that they can be indexed.

## 4.2 Real-World Query Sets

A realistic search scenario requires queries that approximate user needs. Therefore, we created a set of queries based on the Yahoo! Search Query Tiny Sample v1.0 dataset, which contains over four thousand real queries from Yahoo's US query log of January, 2009. Each query in the log is asked by at least three different users and long numbers have been removed for privacy reasons. The query log is provided by the Yahoo! Webscope program[2].

For the entity search task, we selected 50 queries which name an entity explicitly and may also provide some additional context about it, as described in [15]. In the case of the list search track, we hand-picked 50 queries from the Yahoo query log as well as from TrueKnowledge 'recent' queries[3]. The queries describe a closed set of entities, have a relatively small number of possible answers (less than 12) which are unlikely to change.

Although many competitions use queries generated manually by the participants, it is unlikely that those queries are representative of the kinds of entity-based queries used on the Web. For example, queries around religious beliefs are quite a high percentage of queries in real web search engine logs. Therefore, we manually selected queries by randomly selecting from the query logs and then manually checked that at least one relevant answer existed on the current web of linked data.

Table 1 shows examples from the query sets for both tracks. The entire query sets are available for download[4].

| | |
|---|---|
| 08 toyota tundra | gods who dwelt on Mount Olympus |
| Hugh Downs | Arab states of the Persian Gulf |
| MADRID | astronauts who landed on the Moon |
| New England Coffee | Axis powers of World War II |
| PINK PANTHER 2 | books of the Jewish canon |
| concord steel | boroughs of New York City |
| YMCA Tampa | Branches of the US military |
| ashley wagner | continents in the world |
| nokia e73 | standard axioms of set theory |
| bounce city humble tx | manfred von richthofen parents |
| University of York | matt berry tv series |

**Table 1: Examples queries from the Entity Query Set (left) and List Query Set (right).**

[2] `http://webscope.sandbox.yahoo.com/`
[3] `http://www.trueknowledge.com/recent/`
[4] `http://semsearch.yahoo.com/datasets.php`

## 4.3 Crowd-Sourced Relevance Judgments

We used Amazon Mechanical Turk to obtain the relevance assessments. This has been shown to be a reliable method for evaluation purposes, producing a rank ordering of search systems equivalent to the ordering produced by expert human assessors for this task [4]. The human assessors were presented with a simple, human readable rendering of RDF triples about the entity shown as attributes and values in a HTML table, with URIs being truncated to their rightmost hierarchical component. The URI of the entity itself was not shown. The rendering showed a maximum of ten attribute-value pairs with RDF attributes given in the specification and text values in English language being given preference. Based on this presentation and the keyword query, the assessors had to decide on a 3-point scale, whether the entity is irrelevant, somewhat relevant, or relevant. For the List Search track, the workers were presented additionally with a reference list of correct entities in addition to the criteria itself, which was obtained through manual searching by the organizers. This was done as the queries were of such difficulty that many assessors may not know the answers themselves.

First, the top 20 results of all runs for each query were pooled. Despite a validation mechanism in the submission process, we encountered problems with lowercased or N-triple encoded URIs, which required additional manual cleanup. URIs that did not appear as a subject in the data collection were discarded. Each result is evaluated by 5 workers and a majority vote yields the final assessment. The pooled evaluation procedure resulted in 3887 assessments for track 1 and 5675 assessments for track 2, which are available at `http://semsearch.yahoo.com/results.php`. The workers were paid $0.20 per batch of 12 assessments, which took them typically one to two minutes to complete. This results in an hourly wage of $6-$12.

The payment can be rejected for workers who try to game the system. To assure the quality of the assessments, we mixed *gold-win* cases, which are considered perfectly relevant by experts, and *gold-loose* cases, which are considered irrelevant, into a batch of 12 tasks presented to a worker. Thereby, we could estimate the quality of the assessments. All workers missing a considerable amounts of the gold-cases were rejected and their tasks put back into the pool to be done by others. Furthermore, we measured the deviation from the majority and observed such factors as the time to complete a batch. Workers obviously too far off were rejected as well. A lesson learnt here was that workers could choose how many batches they complete, which made it hard to measure the deviations for workers, who completed only few batches. In the future, we will require a minimum number of batches to be completed by each worker before being paid, in order to increase the quality assurance.

# 5. ENTITY SEARCH TRACK EVALUATION

Four teams participated in both tracks. These teams were University of Delaware (**UDel**), Digital Enterprise Research Institute (**DERI**), International Institute of Information Technology Hyderabad (**IIIT Hyd**), and Norwegian University of Science and Technology (**NTNU**). Dhirubhai Ambani Institute of Information and Communication Technology (**DA-IICT**) participated additionally in the List Search Track.

Each team was allowed to enter up to three different submissions ('runs') per track, in order to experiment with different system configurations. A submission is an ordered list of URIs for each query in the *trec format* allowing us to use the *trec_eval* software[5] to compute the metrics. In total, 10 runs were submitted for the Entity Search Track and 11 runs for the List Search Track.

In the following sections, we briefly describe and characterize the systems for each track and report on their performance. Detailed system descriptions are available at the Challenge website[6].

In order to categorize the systems and illustrate their different approaches to the entity search task, two major aspects can be distinguished: (1) the internal model for *entity representation*, and (2) the *retrieval model* applied for matching, retrieval, and ranking. Before, we characterize the systems, we discuss these two major aspects.

### *Entity representation.*

teams used a *quad* having the same subject URI as the representation of an entity. Only DERI deviated from this representation and took all quads having the same subject and their contexts as the representation as the representation of an entity. The applied representations of an entity can be characterized by four aspects, which describe how the specifics of the data are taken into account. The RDF data model makes a distinction between object and datatype properties. Datatype properties can be seen as *attribute-value* pairs, where the value is a literal value, usually a text string. In contrast, object properties are typed *relations* in the form of attribute-object pairs, where the object is the URI identifier of another entity rather than a literal value. Since URIs are used as identifiers, each URI has a *domain name*, which can be seen as one kind of provenance. Another provenance aspect is the *context*, which describes the source of the triple in the BTC dataset. The domain is different from the context because URIs with the same domain can be used in different contexts. Whether these aspects are considered, is illustrated in Table 2 as follows:

- *attribute-value:* Are the attribute-values of the triples used in the entity representation (yes + / no −)?
- *relations:* Are the relations to other entities considered (yes + / no −)? The relations are potentially exploitable for ranking, because they form the data graph by linking to other entities. If this information is not taken into account, the relations usually treated as additional attribute-value pairs.
- *domain:* Is the domain information used (yes + / no −)? Entities of a certain domain are some times boosted, because certain domains are considered a-priori as relevant or of high quality. Often entities from *dbpedia.org* are considered for a-priori boosting.
- *context:* Is the context information included in the entity representation (yes + / no −)? This information can be used as well to favor certain sources.

[5] `http://trec.nist.gov/trec_eval/`
[6] `http://semsearch.yahoo.com`

### *Retrieval model.*

All participating systems used inverted indexes to manage their data. Still, the different approaches can be characterized by three main aspects, although a specific systems could use a combination of them, which are: (1) purely *text based* approaches using a 'bag-of-words' representation of entities and common ranking techniques build on TF/IDF or language models[11]. The main notion of these approaches are term statistics calculated from the text representation. (2) The second type are *structured based* approaches which consider the structure of the data and weighted properties differently. In contrast to the text-based approaches, entities are not seen as flat text corpora, but as structured retrieval units. (3) The third aspect denotes whether the structure of the entire data graph is used to derive query independent scores, e.g. by graph analytics like PageRank. Since this aspect uses the structure for a-priori query scores, we refer to them as 'query-independent structure-based' (Q-I-structured-based) approaches.

Table 2 gives an overview of the systems based on the characteristics introduced above.

## 5.1 Overview of Evaluated Systems

| | | UDel | | DERI | | | NTNU | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Run** | **VO** | **Prox** | **1** | **2** | **3** | **Olav** | **Harald** | **Godfrid** |
| Entity representation | attribute-value | + | + | + | + | + | + | + | + |
| | relations | - | - | - | - | + | - | - | + |
| | domain | + | - | - | + | + | + | + | + |
| | context | - | - | + | + | + | - | - | - |
| Retrieval model | Text-based | + | + | + | + | + | + | + | + |
| | Structure-based | - | - | + | + | + | - | + | + |
| | Q-I-structure | - | - | - | - | + | - | - | - |

**Table 2: Feature overview regarding system internal entity representation and retrieval model**

**UDel:**

*Entity representation:* All quads having the same subject URI constituted one entity. Terms extracted from these quads are simply put into one 'bag-of-words' and indexed as one document.
*Retrieval model:* An axiomatic retrieval function was applied by University of Delaware [8]. For run **UDel-Prox**, query term proximity was added to the model, which favors documents having the query terms within a sliding window of 15 terms. The third run **UDel-VO** promotes entities whose URI has a direct match to a query term.

**DERI:**

*Entity representation:* In contrast to the other systems, the Sindice system from DERI took all quads having the same subject and the same context as the description of an entity. Only entity descriptions comprising more than 3 quads were considered. This entity description is internally represented as a labeled tree data model with an entity node as the root, and subsequent attribute and value nodes. In addition, run **DERI-3** used the entire graph structure, so exploiting the relationships of any given entity when ranking.
*Retrieval model:* BM25MF, an extension of BM25F,

which allows fields to have multiple values was used by Sindice to rank entities for all runs. The second and winning run, **DERI-2**, applied additionally query specific weights, namely query coverage and value coverage. These weights indicate how well the query terms are covered by a root node, respectively value node, in the internal data model. The more query terms are covered by a node, the more weight is contributed to this node. In addition, query independent weights were assigned to attributes, whose URI contain certain keywords, e.g. *label*, *title*, *sameas*, and *name*. Run **DERI-3** used additionally the relations to compute query independent scores based on the graph structure.

**IIIT Hyd:**

*Did not provide a system description.*

**NTNU:**

*Entity representation:* NTNU used the *DBPedia* dataset in addition to the BTC to represent entities. An entity is represented by three sub-models, the first comprises all name variants of this entity in DBPedia, the second considers several attributes from DBPedia for this entity, and the third uses the data from BTC about this entity. On the syntactic level, all triples having the same subject URI were used for the models based on DBPedia. For run **NTNU-Olav**, the model based on the BTC used only literal objects and regarded them as one flat text representation. For the runs **NTNU-Harald** and **NTNU-Godfrid**, the model had two fields, the name field which contained values of attributes that mentioned the name of the entity, while all other attributes were put into the content field.

*Retrieval model:* Mixture language models were used to incorporate the different entity models in the retrieval function, while weights were applied for specific attributes of DBPedia. Run **NTNU-Godfrid** used *sameAs* (an equivalence link on the Semantic Web) relations to propagate scores, in order to rank directly related entities higher.

## 5.2 Entity Search Track Results

The retrieval performance for the submitted runs are given in Table 3. The numbers on precision at cutoffs (P5, P10) give an impression about the top of the returned result lists. Mean Average Precision (MAP) takes the entire ranked list into account and is based on the complete assessment pool. On average, there are 9.4 relevant entities per query with a standard deviation of 11. For four queries no system could deliver relevant entities, which shows that some queries were really hard. These were queries with number *q18*, *q24*, *q25* and *q29*, e.g. *q25*: "holland bakery jakarta".

### *Discussion of the Entity Search Track.*

The semantic search task of finding entities in an large RDF graph has been addressed by a spectrum of different approaches in this challenge as shown by the diversity of the results. The basis for most system are still the well known Information Retrieval techniques, which yields acceptable results. However, the winning system from DERI is a specialized system, which adapted IR methods and tailored them to RDF. The key feature for success, shared by the two top ranked systems in this years challenge, is to take the proximity or coverage of query terms on individual attribute values into account. This is a consequent development step over last year's challenge, where weighting properties individual was the key feature for success. The general observation is that considering the particular pieces of the structured data yields higher performance over unstructured, text-based retrieval shows that search can benefit from more structure.

| Participant | Run | P10 | P5 | **MAP** |
|---|---|---|---|---|
| DERI | 2 | 0.260 | 0.332 | **0.2346** |
| UDel | Prox | 0.260 | 0.337 | **0.2167** |
| NTNU | Harald | 0.222 | 0.280 | **0.2072** |
| NTNU | Godfrid | 0.224 | 0.272 | **0.2063** |
| NTNU | Olav | 0.220 | 0.276 | **0.2050** |
| UDel | VO | 0.194 | 0.248 | **0.1858** |
| DERI | 1 | 0.218 | 0.292 | **0.1835** |
| DERI | 3 | 0.188 | 0.252 | **0.1635** |
| IIIT Hyd | 1 | 0.130 | 0.148 | **0.0876** |
| IIIT Hyd | 2 | 0.142 | 0.132 | **0.0870** |

**Table 3: Results of the Entity Search Track.**

Similar to last year, one of the main and promising features of the RDF data model, namely the ability to express and type the relations between entities was only used by one run from DERI, which did not exceed the other runs. Whether relations are actually not helpful for entity search on large scale datasets or whether the usage of the relations is not yet understood remains to be investigated in the future. The List Search Track was designed with the intention in mind to get the systems to consider the relations as well. How the systems addressed this task is described in the next section.

# 6. LIST SEARCH TRACK EVALUATION

In general the teams participated with the same systems in the List Search Track and adapted them only slightly to this new task, although the most high-performing system was specially designed for the List Track. The adaptions are mostly on query analysis and interpretation, because the queries were not just keywords but more complex descriptions in natural language, as described in Section4.2. The modifications as well as the additional system are described in the next section followed by the results for this track. The modifications as well as the additional system are described in the next section followed by the results for this track.

## 6.1 Overview of Evaluated Systems

**Delaware:**

The team from Delaware applied an NLP parser to process the queries for run **UDelRun1**, in order to find the target type of the entities. Only entities belonging to this type were considered as results. For the runs **UDelRun2** and **UDelRun3** the type information was manually expanded, because the automatic processing failed in some cases. Instead of the axiomatic retrieval function, model-based relevance feedback was applied for run **UDelRun3** [17].

**DERI:**

DERI participated with an identical system configuration in the List Search Track.

**NTNU:**

NTNU participated with a system especially designed for this track. The system used only the Wikipedia

dataset and mapped the results to entities in the BTC collection. The queries were analyzed and potentially reformulated using the Wikipedia Miner software [12], in order to find the primary entity of the query. The query was run against an index of Wikipedia abstracts to get a candidate list of Wikipedia articles. The outgoing links from these articles were expanded and the resulting articles were also added to the candidate list. Scores are added if an article occurs multiple times and articles with a direct relation to the principal entity are boosted. In contrast to run **NTNU-1**, the runs **NTNU-2** and **NTNU-3** used an additional boosting for articles belonging to a Wikipedia set that had more than a certain fraction of its set of members in the candidate list. Run **NTNU-3** also applied an additional boost based on *sameAs* links.

**DA-IICT:**

The system by DA-IICT used a text-based approach build on Terrier [14] which favored entities according to the number of query terms present in their textual description. Due to data loss, the queries were only run against a part of the BTC data collection.

## 6.2 List Search Track Results

The retrieval performance for the submitted runs are shown in Table 4. The metrics were computed the same ways as for the Entity Track. There are on average 13 relevant entities per query with a standard deviation of 12.8. The participating systems could not find relevant entities for 6 queries. These were the queries with numbers *q15*, *q23*, *q27*, *q28*, *q45* and *q48*, for example *q15: "henry ii's brothers and sisters"*.

| Participant | Run | P10 | P5 | **MAP** |
|---|---|---|---|---|
| NTNU | 3 | 0.354 | 0.356 | 0.2790 |
| NTNU | 2 | 0.348 | 0.372 | 0.2594 |
| NTNU | 1 | 0.204 | 0.200 | 0.1625 |
| DERI | 1 | 0.210 | 0.220 | 0.1591 |
| DERI | 3 | 0.186 | 0.216 | 0.1526 |
| DERI | 2 | 0.192 | 0.216 | 0.1505 |
| UDel | 1 | 0.170 | 0.200 | 0.1079 |
| UDel | 2 | 0.162 | 0.152 | 0.0999 |
| IIIT Hyd | 1 | 0.072 | 0.076 | 0.0328 |
| IIIT Hyd | 2 | 0.072 | 0.076 | 0.0328 |
| DA-IICT | 1 | 0.014 | 0.012 | 0.0050 |

**Table 4: Results of the List Search Track.**

### *Discussion of the List Search Track.*

The List Search Track proved to be a hard task and may require different techniques compared to the Entity Search Track. Since this track was new, most teams participated with their systems built for the Entity Search Track and adapted to the task mainly by analyzing and interpreting the query. Still, the performances show that solutions can be delivered, although there is obviously room for improvement. The winning system by NTNU did not use the BTC data collection, but was built on the Wikipedia corpus and exploited the links between articles. Obviously, the plain links between articles are a valuable resource for search. Ideally, such algorithms could eventually be adopted to more general-purpose RDF structured data outside that of Wikipedia.

# 7. CONCLUSIONS

The Semantic Search Challenge started in 2010 with the task of (named) entity retrieval from RDF data crawled from the Web. Though this task is seemingly simple, because the query contains the name of the entity, it features many of the problems in semantic search, including the potential ambiguity of short-form queries, the varying degrees of relevance by which an entity can be related to the one named in the query and the general quality issues inherent to Web data. The List Search Track introduced this year presented an even harder problem, i.e. queries that don't explicitly name an entity, but rather describe the set of matching entities.

The general direction of our work will continue toward exploring search tasks of increasing difficulty. In addition, there are a number of open questions that may impact the end-user benefits of semantic search engines and would still need to be investigated. For example, the retrieval engines above do not attempt to remove duplicates, and may return different, redundant descriptions of the same entity multiple times. A semantic search engine should remove such duplicates or merge them. Similarly, the user experience is largely impacted by the explanations given by the search engines. Similar to how current text search engines generate summaries and highlight keyword matches, a semantic search engine should attempt to summarize information from an RDF graph and highlight why a particular result is an answer to the user's query.

# 8. ACKNOWLEDGMENTS

We acknowledge Yahoo! Labs for hosting the Semantic Search Challenge 2011 website and sponsoring the prizes. The costs of the evaluation have been sponsored by the European SEALS project, `http://www.seals-project.eu`.

# 9. ADDITIONAL AUTHORS

Thanh Tran Duc, Institute AIFB, Karlsruhe Institute of Technology, 76128 Karlsruhe, Germany ducthanh.tran@kit.edu

# 10. REFERENCES

[1] K. Balog, P. Serdyukov, and A. de Vries. Overview of the trec 2010 entity track. In *TREC 2010 Working Notes*, 2010.

[2] T. Berners-Lee and M. Fischetti. *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. Harper San Francisco, 1999.

[3] C. Bizer and P. Mika. The semantic web challenge, 2009. *Journal of Web Semantics*, 8(4):341, 2010.

[4] R. Blanco, H. Halpin, D. M. Herzig, P. Mika, J. Pound, H. S. Thompson, and D. T. Tran. Repeatable and reliable search system evaluation using crowd-sourcing. In *SIGIR*. ACM, 2011.

[5] C. Cleverdon and M. Kean. Factors Determining the Performance of Indexing Systems, 1966.

[6] J. Coffman and A. C. Weaver. A framework for evaluating database keyword search strategies. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 729–738, New York, NY, USA, 2010. ACM.

[7] G. Demartini, T. Iofciu, and A. P. De Vries. Overview of the inex 2009 entity ranking track. In *Proceedings of the Focused retrieval and evaluation, and 8th international conference on Initiative for the evaluation of XML retrieval*, INEX'09, pages 254–264, Berlin, Heidelberg, 2010. Springer-Verlag.

[8] H. Fang and C. Zhai. An exploration of axiomatic approaches to information retrieval. In *SIGIR*, pages 480–487, 2005.

[9] H. Halpin, D. M. Herzig, P. Mika, R. Blanco, J. Pound, H. S. Thompson, and D. T. Tran. Evaluating Ad-Hoc Object Retrieval. In *Proceedings of the International Workshop on Evaluation of Semantic Technologies (IWEST 2010)*. 9th International Semantic Web Conference (ISWC2010), 2010.

[10] *Proceedings of the Second International Workshop on Keyword Search on Structured Data, KEYS 2010, Indianapolis, Indiana, USA, June 6, 2010*. ACM, 2010.

[11] C. D. Manning, P. Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[12] D. Milne and I. H. Witten. An open-source toolkit for mining wikipedia. In *Proc. New Zealand Computer Science Research Student Conf.*, volume 9, 2009.

[13] E. Oren, R. Delbru, M. Catasta, R. Cyganiak, H. Stenzhorn, and G. Tummarello. Sindice.com: a document-oriented lookup index for open linked data. *IJMSO*, 3(1):37–52, 2008.

[14] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and C. Lioma. Terrier: A High Performance and Scalable Information Retrieval Platform. In *Proceedings of ACM SIGIR'06 Workshop on Open Source Information Retrieval (OSIR 2006)*, 2006.

[15] J. Pound, P. Mika, and H. Zaragoza. Ad-hoc Object Ranking in the Web of Data . In *Proceedings of the WWW*, pages 771–780, Raleigh, United States of America, 2010.

[16] C. Unger, P. Cimiano, V. Lopez, and E. Motta. QALD-1 Open Challenge, 2011. http://www.sc.cit-ec.uni-bielefeld.de/sites/www.sc.cit-ec.uni-bielefeld.de/files/sharedtask.pdf.

[17] C. Zhai and J. D. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *CIKM*, pages 403–410, 2001.