

Entity Ranking in Wikipedia

Anne-Marie Vercoustre, James A. Thom, Jovan Pehcevski

▶ **To cite this version:**

Anne-Marie Vercoustre, James A. Thom, Jovan Pehcevski. Entity Ranking in Wikipedia. the 23rd Annual ACM Symposium on Applied Computing, Mar 2008, Fortaleza, Brazil. inria-00189149

HAL Id: inria-00189149
https://inria.hal.science/inria-00189149

Submitted on 20 Nov 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Entity Ranking in Wikipedia

Anne-Marie Vercoustre
INRIA
Rocquencourt, France
anne-marie.vercoustre@inria.fr

James A. Thom
RMIT University
Melbourne, Australia
james.thom@rmit.edu.au

Jovan Pehcevski
INRIA
Rocquencourt, France
jovan.pehcevski@inria.fr

inria-00189149, version 1 - 20 Nov 2007

ABSTRACT

The traditional entity extraction problem lies in the ability of extracting named entities from plain text using natural language processing techniques and intensive training from large document collections. Examples of named entities include organisations, people, locations, or dates. There are many research activities involving named entities; we are interested in entity ranking in the field of information retrieval. In this paper, we describe our approach to identifying and ranking entities from the INEX Wikipedia document collection. Wikipedia offers a number of interesting features for entity identification and ranking that we first introduce. We then describe the principles and the architecture of our entity ranking system, and introduce our methodology for evaluation. Our preliminary results show that the use of categories and the link structure of Wikipedia, together with entity examples, can significantly improve retrieval effectiveness.

Categories and Subject Descriptors: H.3 [Information Storage and Retrieval]: H.3.3 Information Search and Retrieval

Keywords: Entity Ranking, XML Retrieval, Test collection

1. INTRODUCTION

Information systems contain references to many named entities. In a well-structured database system it is exactly clear what are references to named entities, whereas in semi-structured information sources (such as web pages) it is harder to identify them within a text. An entity could be, for example, an organisation, a person, a location, or a date. Because of the importance of named entities, several very active and related research areas have emerged in recent years, including: entity extraction/tagging from texts, entity reference solving, entity disambiguation, question-answering, expert search, and entity ranking (also known as entity retrieval).

The traditional entity extraction problem is to extract named entities from plain text using natural language processing techniques or statistical methods and intensive training from large collections [10, 25]. Benchmarks for evaluation of entity extraction have been performed for the Message Understanding Conference (MUC) [26] and for the Automatic Content Extraction (ACE) program [21]. Training is done on a large number of examples in order to identify extraction patterns (rules). The goal is to eventually tag those entities and use the tag names to support future information retrieval.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
SAC'08 March 16-20, 2008, Fortaleza, Ceará, Brazil
Copyright 2008 ACM 978-1-59593-753-7/08/0003 ...$5.00.

In the context of large collections such as the web or Wikipedia, it is not possible, nor even desirable, to tag in advance all the entities in the collection, although many occurrences of named entities in the text may be used as anchor text for sources of hypertext links. Instead, since we are dealing with semi-structured documents (HTML or XML), we could exploit the explicit document structure to infer effective extraction patterns or algorithms.

The goal of *entity ranking* is to retrieve entities as answers to a query. The objective is not to tag the names of the entities in documents but rather to get back a list of the relevant entity names (possibly each entity with an associated description). For example, the query "European countries where I can pay with Euros" [11] should return a list of entities (or pages) representing relevant countries, and not a list of entities about the Euro and similar currencies.

The Initiative for the Evaluation of XML retrieval (INEX) has a new track on entity ranking [12], using Wikipedia as its document collection. This track proposes two tasks: a task where the category of the expected entity answers is provided; and a task where a few (two or three) examples of the expected entity answers are provided. The inclusion of target categories (in the first task) and example entities (in the second task) makes these quite different tasks from full-text retrieval. The combination of the query and example entities (in the second task) makes it quite different from an application such as Google Sets[1] where only entity examples are provided.

In this paper, we identify some important principles for entity ranking that we incorporate into an architecture which allows us to tune, evaluate, and improve our approach as it develops. Our entity ranking approach is based on three ideas: (1) using full-text similarity with the query, (2) using popular links (from highly scored pages), and (3) using category similarity with the entity examples.

2. RELATED WORK

Our entity ranking approach gets its inspiration from wrapping technology, entity extraction, the use of ontologies for entity extraction or entity disambiguation, and link analysis.

Wrappers

A wrapper is a tool that extracts information (entities or values) from a document, or a set of documents, with a purpose of reusing information in another system. A lot of research has been carried out in this field by the database community, mostly in relation to querying heterogeneous databases [1, 16, 24, 28]. More recently, wrappers have also been built to extract information from web pages with different applications in mind, such as product comparison, reuse of information in virtual documents, or build-

[1] http://labs.google.com/sets

ing experimental data sets. Most web wrappers are either based on scripting languages [24, 28] that are very close to current XML query languages, or use wrapper induction [1, 16] that learn rules for extracting information.

To prevent wrappers breaking over time without notice when pages change, Lerman et al. [17] propose using machine learning for wrapper verification and re-induction. Rather than repairing a wrapper over changes in the web data, Callan and Mitamura [6] propose generating the wrapper dynamically — that is at the time of wrapping, using data previously extracted and stored in a database. The extraction rules are based on heuristics around a few pre-defined lexico-syntactic HTML patterns such as lists, tables, and links. The patterns are weighted according to the number of examples they recognise; the best patterns are used to dynamically extract new data.

Our system for entity ranking also works dynamically, at query time instead of at wrapping time. We also use weighting algorithms based on links that are well represented in web-based collections, as well as knowledge of categories, a specific Wikipedia feature.

Entity extraction

Recent research in named entity extraction has developed approaches that are not language dependant and do not require lots of linguistic knowledge. McNamee and Mayfield [20] developed a system for entity extraction based on training on a large set of very low level textual patterns found in tokens. Their main objective was to identify entities in multilingual texts and classify them into one of four classes (location, person, organisation, or "others"). Cucerzan and Yarowsky [9] describe and evaluate a language-independent bootstrapping algorithm based on iterative learning and re-estimation of contextual and morphological patterns. It achieves competitive performance when trained on a very short labelled name list.

Using ontology for entity extraction

Other approaches for entity extraction are based on the use of external resources, such as an ontology or a dictionary. Popov et al. [23] use a populated ontology for entity extraction, while Cohen and Sarawagi [7] exploit a dictionary for named entity extraction. Tenier et al. [27] use an ontology for automatic semantic annotation of web pages. Their system firstly identifies the syntactic structure that characterises an entity in a page, and then uses subsumption to identify the more specific concept to be associated with this entity.

Using ontology for entity disambiguation

Hassell et al. [14] use a "populated ontology" to assist in disambiguation of entities, such as names of authors using their published papers or domain of interest. They use text proximity between entities to disambiguate names (e.g. organisation name would be close to author's name). They also use text co-occurrence, for example for topics relevant to an author. So their algorithm is tuned for their actual ontology, while our algorithm is more based on the the categories and the structural properties of the Wikipedia.

Cucerzan [8] uses Wikipedia data for named entity disambiguation. He first pre-processed a version of the Wikipedia collection (September 2006), and extracted more than 1.4 millions entities with an average of 2.4 surface forms by entities. He also extracted more than one million (entities, category) pairs that were further filtered down to 540 thousand pairs. Lexico-syntactic patterns, such as titles, links, paragraphs and lists, are used to build coreferences of entities in limited contexts. The knowledge extracted from Wikipedia is then used for improving entity disambiguation in the context of web and news search.

Link Analysis (PageRank and HITS)

Most information retrieval (IR) systems use statistical information concerning the distribution of the query terms to calculate the query-document similarity. However, when dealing with hyperlinked environments such as the web or Wikipedia, link analysis is also important. PageRank and HITS are two of the most popular algorithms that use link analysis to improve web search performance.

PageRank, an algorithm proposed by Brin and Page [5], is a link analysis algorithm that assigns a numerical weighting to each page of a hyperlinked set of web pages. The idea of PageRank is that a web page is a good page if it is popular, that is if many other (also preferably popular) web pages are referring to it.

In HITS (Hyperlink Induced Topic Search), *hubs* are considered to be web pages that have links pointing to many *authority* pages [15]. However, unlike PageRank where the page scores are calculated independently of the query by using the complete web graph, in HITS the calculation of hub and authority scores is query-dependent; here, the so-called *neighbourhood graph* includes not only the set of top-ranked pages for the query, but it also includes the set of pages that either point to or are pointed to by these pages.

We use the idea behind PageRank and HITS in our system; however, instead of counting every possible link referring to an entity page in the collection (as with PageRank), or building a neighbourhood graph (as with HITS), we only consider pages that are pointed to by a selected number of top-ranked pages for the query. This makes our link ranking algorithm query-dependent (just like HITS), allowing it to be dynamically calculated at query time.

3. INEX ENTITY RANKING TRACK

The INEX Entity ranking track was proposed as a new track in 2006, but will only start in 2007. It will use the Wikipedia XML document collection (described in the next section) that has been used by various INEX tracks in 2006 [19]. Two tasks are planned for the INEX Entity ranking track in 2007 [12]:

task1: entity ranking, where the aim is to retrieve entities of a given category satisfying a topic described by a few query terms;

task2: list completion, where given a topic text and a number of examples, the aim is to complete this partial list of answers.

Figure 1 shows an example INEX entity ranking topic; the `title` field contains the query terms, the `description` provides a natural language summary of the information need, and the `narrative` provides an explanation of what makes an entity answer relevant. In addition, the `entities` field provides a few of the expected entity answers for the topic (task 2), while the `categories` field provides the category of the expected entity answers (task 1).

4. THE INEX WIKIPEDIA CORPUS

Wikipedia is a well known web-based, multilingual, free content encyclopedia written collaboratively by contributors from around the world. As it is fast growing and evolving it is not possible to use the actual online Wikipedia for experiments. Denoyer and Gallinari [13] have developed an XML-based corpus founded on a snapshot of the Wikipedia, which has been used by various INEX tracks in 2006. It differs from the real Wikipedia in some respects (size, document format, category tables), but it is a very realistic approximation. Specifically, the INEX Wikipedia XML document corpus retains the main characteristics of the online version, although they have been implemented through XML tags instead of

```
<inex_topic>
<title>
European countries where I can pay with Euros
</title>
<description>
I want a list of European countries where
I can pay with Euros.
</description>
<narrative>
Each answer should be the article about a specific
European country that uses the Euro as currency.
</narrative>
<entities>
   <entity id="10581">France</entity>
   <entity id="11867">Germany</entity>
   <entity id="26667">Spain</entity>
</entities>
<categories>
   <category id="61">countries<category>
</categories>
</inex_topic>
```

Figure 1: Example INEX 2007 entity ranking topic

the initial HTML tags and the native Wikipedia structure. The corpus is composed of 8 main collections, corresponding to 8 different languages. The INEX 2007 Entity ranking track will use the 4.6 Gigabyte English sub-collection which contains 659,388 articles.

4.1 Entities in Wikipedia

In Wikipedia, an entity is generally associated with an article (a Wikipedia page) describing this entity. Nearly everything can be seen as an entity with an associated page, including countries, famous people, organisations, places to visit, and so forth.

The entities have a name (the name of the corresponding page) and a unique ID in the collection. When mentioning such an entity in a new Wikipedia article, authors are encouraged to link at least the first occurrence of the entity name to the page describing this entity. This is an important feature as it allows to easily locate potential entities, which is a major issue in entity extraction from plain text. Consider the following extract from the Euro page.

> "The **euro** ... is the official currency of the Eurozone (also known as the Euro Area), which consists of the European states of Austria, Belgium, Finland, France, Germany, Greece, Ireland, Italy, Luxembourg, the Netherlands, Portugal, Slovenia and Spain, and will extend to include Cyprus and Malta from 1 January 2008."

All the underlined words (hypertext links that are usually highlighted in another colour by the browser) can be seen as occurrences of entities that are each linked to their corresponding pages. In this extract, there are 18 entity references of which 15 are country names; these countries are all "European Union member states", which brings us to the notion of category in Wikipedia.

4.2 Categories in Wikipedia

Wikipedia also offers categories that authors can associate with Wikipedia pages. New categories can also be created by authors, although they have to follow Wikipedia recommendations in both creating new categories and associating them with pages. For example, the Spain page is associated with the following categories: "Spain", "European Union member states", "Spanish-speaking countries", "Constitutional monarchies" (and some other Wikipedia administrative categories). There are 113,483 categories in the INEX Wikipedia XML collection, which are organised in a graph of categories. Each page can be associated with many categories (2.28 as an average). Some properties of Wikipedia categories (explored in more detail by Yu et al. [30]) include:

- a category may have many sub-categories and parent categories;
- some categories have many associated pages (i.e. large extension), while others have smaller extension;
- a page that belongs to a given category extension generally does not belong to its ancestors' extension;
- the sub-category relation is not always a subsumption relationship; and
- there are cycles in the category graph.

When searching for entities it is natural to take advantage of the Wikipedia categories since they give hints on whether the retrieved entities are of the expected type. For example, if looking for 'author' entities, pages associated with the category "novelists" may be more relevant than pages associated with the category "books".

5. OUR APPROACH

In this work, we are addressing the task of ranking entities in answer to a query supplied with a few examples (task 2). Our approach is based on the following principles for entity answer pages. A good page:

- answers the query (or a query extended with the examples),
- is associated with a category close to the categories of the entity examples (we use a similarity function between the categories of a page and the categories of the examples),
- is pointed to by a page answering the query (this is an adaptation of the HITS [15] algorithm to the problem of entity ranking; we refer to it as a linkrank algorithm), and
- is pointed to by contexts with many occurrences of the entity examples. We currently use the full page as the context when calculating the scores in our linkrank algorithm. Smaller contexts such as paragraphs, lists, or tables have been used successfully by others [18].

We have built a system based on the above principles, where candidate pages are ranked by combining three different scores: a linkrank score, a category score, and the initial search engine similarity score. We use Zettair,[2] a full-text search engine developed by RMIT University, which returns pages ranked by their similarity score to the query. We use the Okapi BM25 similarity measure as it was effective on the INEX Wikipedia collection [2].

Our system involves several modules for processing a query, submitting it to the search engine, applying our entity ranking algorithms, and finally returning a ranked list of entities, including:

the topic module takes an INEX topic as input (as the topic example shown in Figure 1) and generates the corresponding Zettair query and the list of entity examples (as an option, the example entities may be added to the query);

the search module sends the query to Zettair and returns a list of ranked Wikipedia pages (typically 1500); and

the link extraction module extracts the links from a selected number of highly ranked pages together with the XML paths of the links (we discard external links and internal collection links that do not refer to existing pages in the collection).

Using the pages found by these modules, we calculate a global score for each page (see 5.4) as a linear combination of the normalised scores coming out from the following three functions:

[2]http://www.seg.rmit.edu.au/zettair/

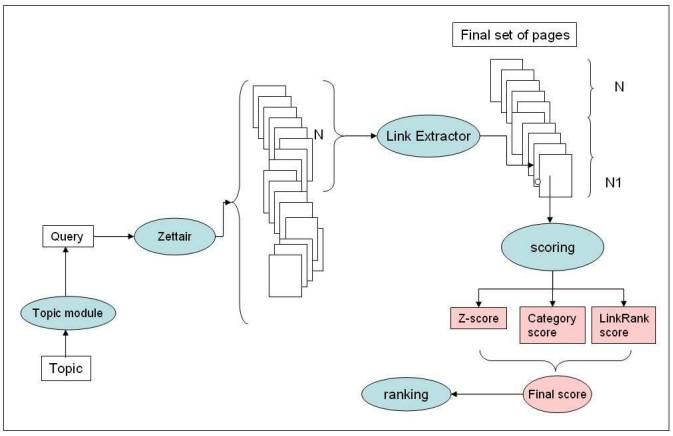


Figure 2: Process for Entity ranking

- **the linkrank function**, which calculates a weight for a page based (among other things) on the number of links to this page (see 5.1);
- **the category similarity function**, which calculates a weight for a page based on the similarity of the page categories with those of the entity examples (see 5.2); and
- **the full-text IR function**, which calculates a weight for a page based on its initial Zettair score (see 5.3).

The overall process for entity ranking is shown in Figure 2. The architecture provides a general framework for evaluating entity ranking which allows for replacing some modules by more advanced modules, or by providing a more efficient implementation of a module. It also uses an evaluation module (not shown in the figure) to assist in tuning the system by varying the parameters and to globally evaluate the entity ranking approach.

5.1 LinkRank score

The linkrank function calculates a score for a page, based on the number of links to this page, from the first N pages returned by the search engine in response to the query. We carried out some experiments with different values of N and found that N=20 was an acceptable compromise between performance and discovering more potentially good entities. The linkrank function can be implemented in a variety of ways: by weighting pages that have more links referring to them from higher ranked pages (the initial N pages), or from pages containing larger number of entity examples, or a combination of the two. We have implemented a very basic linkrank function that, for a target entity page t, takes into account the Zettair score of the referring page $z(p_r)$, the number of distinct entity examples in the referring page $\#ent(p_r)$, and the number of reference links to the target page $\#links(p_r, t)$:

$$S_L(t) = \sum_{r=1}^{N} z(p_r) \cdot g(\#ent(p_r)) \cdot f(\#links(p_r, t)) \quad (1)$$

where $g(x) = x + 0.5$ (we use 0.5 to allow for cases where there are no entity examples in the referring page) and $f(x) = x$ (as there is at least one reference link to the target page).

5.2 Category similarity score

There has been a lot of research on similarity between concepts of two ontologies, especially for addressing the problem of mapping or updating ontologies [3]. Similarity measures between concepts of the same ontology cannot be applied directly to Wikipedia categories, mostly because the notion of sub-categories in Wikipedia is not a subsumption relationship. Another reason is that categories in Wikipedia do not form a hierarchy (or a set of hierarchies) but a graph with potential cycles. Therefore tree-based similarities [4] either cannot be used or their applicability is limited.

However, the notions of ancestors, common ancestors, and shorter paths between categories can still be used, which may allow us to define a distance between the set of categories associated with a given page, and the set of categories associated with the entity examples. We use a very basic similarity function that is the ratio of common categories between the set of categories associated with the target page $\mathsf{cat}(t)$ and the union of the categories associated with the entity examples $\mathsf{cat}(E)$:

$$S_C(t) = \frac{|\mathsf{cat}(t) \cap \mathsf{cat}(E)|}{|\mathsf{cat}(E)|} \quad (2)$$

5.3 Z score

The Z score assigns the initial Zettair score to a target page. If the target page does not appear in the list of 1500 ranked pages returned by Zettair, then its Z score is zero:

$$S_Z(t) = \begin{cases} z(t) & \text{if page } t \text{ was returned by Zettair} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

5.4 Global score

The global score $S(t)$ for a target entity page is calculated as a linear combination of three normalised scores, the linkrank score $S_L(t)$, the category similarity score $S_C(t)$, and the Z score $S_Z(t)$:

$$S(t) = \alpha S_L(t) + \beta S_C(t) + (1 - \alpha - \beta) S_Z(t) \quad (4)$$

where α and β are parameters that can be tuned. Some special cases let us evaluate the effectiveness of each module in our system: where only the linkrank score is used ($\alpha = 1$, $\beta = 0$); where only the category score is used ($\alpha = 0$, $\beta = 1$); and where only the Z score is used[3] ($\alpha = 0$, $\beta = 0$). More combinations of the two parameters are explored in the training phase of our system.

6. EXPERIMENTAL RESULTS

6.1 Evaluation Methodology

There is no existing set of topics with assessments for entity ranking, although such a set will be developed for the INEX entity ranking track in 2007. So we developed our own test collection based on a selection of topics from the INEX 2006 ad hoc track. We chose 27 topics from the INEX 2006 ad hoc track that we considered were of an "entity ranking" nature. For each page that had been assessed as containing relevant information, we reassessed whether or not it was an entity answer, that is whether it *loosely* belonged to a category of entity we had *loosely* identified as being the target of the topic. We did not require that the answers should strictly belong to a particular category in the Wikipedia. If there were example entities mentioned in the original topic, then these were usually used as entity examples in the entity topic. Otherwise, a selected number (typically 2 or 3) of entity examples were chosen somewhat arbitrarily from the relevance assessments.

[3]This is not the same as the plain Zettair score, as apart from the highest N pages returned by Zettair, the remaining N1 entity answers are all generated by extracting links from these pages.

Table 1: Mean average precision scores for runs using 66 possible α–β combinations, obtained on the 11 INEX 2006 training topics. Queries sent to Zettair include only terms from the topic title (Q). The MAP score of the plain Zettair run is 0.1091. The numbers in italics show the scores obtained for each of the three individual modules. The best performing MAP score is shown in bold.

	Beta (β)										
Alpha (α)	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.0	*0.1189*	0.1374	0.1688	0.1891	0.2190	0.2158	0.2241	0.2295	0.2424	0.2505	*0.2382*
0.1	0.1316	0.1479	0.1917	0.2041	0.2299	0.2377	0.2562	0.2669	0.2707	0.2544	
0.2	0.1428	0.1644	0.1897	0.2279	0.2606	0.2655	0.2795	0.2827	0.2602		
0.3	0.1625	0.1893	0.2058	0.2383	0.2703	0.2766	**0.2911**	0.2631			
0.4	0.1774	0.1993	0.2220	0.2530	0.2724	0.2822	0.2638				
0.5	0.1878	0.2075	0.2279	0.2517	0.2762	0.2623					
0.6	0.1979	0.2153	0.2441	0.2460	0.2497						
0.7	0.2011	0.2187	0.2342	0.2235							
0.8	0.2016	0.2073	0.2006								
0.9	0.1939	0.1843									
1.0	*0.1684*										

We used the first 9 of the 27 topics in our training set, to which we added two more topics created by hand from the original track description (one of these extra topics is the Euro example in Figure 1). The remaining 18 topics were our test set.

We use MAP (mean average precision) as our primary method of evaluation, but also report some results with other measures which are typically used to evaluate the retrieval performance of IR systems [29]. We first remove the the entity examples both from the list of answers returned by each system and from the relevance assessments (as the task is to find entities other than the examples provided). We calculate precision at rank r as follows:

$$P[r] = \frac{\sum_{i=1}^{r} rel(i)}{r} \quad (5)$$

where $rel(i) = 1$ if the i^{th} article in the ranked list was judged as a relevant entity, 0 otherwise. Average precision is calculated as the average of $P[r]$ for each relevant entity retrieved (that is at natural recall levels); if a system does not retrieve a particular relevant entity, then the precision for that entity is assumed to be zero. MAP is the mean value of the average precisions over all the topics in the training (or test) data set.

We also report on several alternative measures: mean of $P[1]$, $P[5]$, $P[10]$ (mean precision at top 1, 5 or 10 entities returned), mean R-precision (R-precision for a topic is the $P[R]$, where R is the number of entities that have been judged relevant for the topic).

6.2 Training data set (11 topics)

We used the training data set to determine suitable values for the parameters α and β. We varied α from 0 to 1 in steps of 0.1, and for each value of α, we varied β from 0 to $(1 - \alpha)$ in steps of 0.1. We observe in Table 1 that the highest MAP (0.2911) on the 11 topics is achieved for $\alpha = 0.3$ and $\beta = 0.6$. We also tried training using mean R-precision instead of MAP as our evaluation measure where we observed somewhat different optimal values for the two parameters: $\alpha = 0.1$ and $\beta = 0.8$. A reason for this is the relatively small number of topics used in the training data set. We would expect the optimal parameter values obtained by MAP and R-precision to converge if many more training topics are used.

On the training data, we also experimented with adding the names of the example entities to the query sent to Zettair. However this generally performed worse, both for the plain Zettair run and the runs using various score combinations, and so it would require a more detailed per-topic analysis in order to investigate why this occurs. Accordingly, for the test collection we use queries that only include terms from the topic title, and consider the plain Zettair run as a baseline while comparing our entity ranking approaches.

Table 2: Performance scores for runs obtained with different evaluation measures, using the 18 INEX 2006 test topics. Queries sent to Zettair include only terms from the topic title (Q). The best performing scores are shown in bold.

	P[r]				
Run	1	5	10	***R-prec***	***MAP***
Zettair	0.2778	0.3000	0.2722	0.2258	0.2023
α0.0–β0.0	0.2778	0.3000	0.2722	0.2363	0.2042
α0.0–β1.0	**0.5556**	0.4111	0.3444	0.3496	0.3349
α1.0–β0.0	0.0556	0.1556	0.1278	0.1152	0.1015
α0.3–β0.6	0.5000	0.4444	0.3667	0.3815	0.3274
α0.1–β0.8	**0.5556**	**0.5333**	**0.4222**	**0.4337**	**0.3899**

6.3 Test data set (18 topics)

In these experiments, we designed runs to compare six entity ranking approaches using the 18 topics in the test data set:

- full-text retrieval using Zettair (as a baseline)
- link extraction and re-ranking using the Z score (S_Z)
- link extraction and re-ranking using the category score (S_C)
- link extraction and re-ranking using the linkrank score (S_L)
- link extraction and re-ranking using two global scores:
  - $(0.3 * S_L + 0.6 * S_C + 0.1 * S_Z)$
  - $(0.1 * S_L + 0.8 * S_C + 0.1 * S_Z)$

The results for these six runs are shown in Table 2. We observe that the best entity ranking approach is the one that places most of the weight on the category score S_C (run α0.1–β0.8). With both MAP and R-precision, this run performs significantly better ($p < 0.05$) than the plain Zettair full-text retrieval run and the other four runs that use various score combinations in the re-ranking. The run that uses the category score in the re-ranking performs the best among the three runs that represent the three individual modules; however, statistically significant performance difference ($p < 0.05$) is only observed when comparing this run to the worst performing run (α1.0–β0.0) which uses only the linkrank score in the re-ranking.

These results show that the global score (the combination of the three individual scores), optimised in a way to give more weight on the category score, brings the best value in retrieving the relevant entities for the INEX Wikipedia document collection.

7. CONCLUSION AND FUTURE WORK

We have presented our entity ranking approach for the INEX Wikipedia XML document collection which is based on exploiting the interesting structural and semantic properties of the collection. We have shown in our preliminary evaluations that simple use of the categories and the link structure of Wikipedia, together with the entity examples from the topic, can significantly improve the entity ranking performance compared to a full-text retrieval engine.

Our current implementation uses very simple linkrank and category similarity functions and offers room for improvement.

To improve the linkrank function, we plan to narrow the context around the entity examples. We expect relevant entities to frequently co-occur with the example entities in lists. The narrower context could be defined either by fixed XML elements (such as paragraphs, lists, or tables) or it could be determined dynamically. To determine it dynamically, we plan to identify *coherent retrieval elements* adapting earlier work by Pehcevski et al. [22] to identify the element contexts that are most likely to contain lists.

To improve the category similarity function, we plan to take into account the notion of existing sub-categories and parent categories found in Wikipedia.

We will also be participating in the INEX 2007 entity ranking track, which we expect would enable us to test our approach using a larger set of topics and compare it against alternative approaches.

Acknowledgements This work was undertaken while James Thom was visiting INRIA in 2007.

REFERENCES

[1] B. Adelberg and M. Denny. Nodose version 2.0. In *SIGMOD '99: Proc. 1999 ACM SIGMOD international conference on Management of data*, pages 559–561, Philadelphia, PA, 1999.

[2] D. Awang Iskandar, J. Pehcevski, J. A. Thom, and S. M. M. Tahaghoghi. Social media retrieval using image features and structured text. In *Comparative Evaluation of XML Information Retrieval Systems: 5th Workshop of the INitiative for the Evaluation of XML Retrieval, INEX 2006*, volume 4518 of *LNCS*, pages 358–372, 2007.

[3] E. Blanchard, M. Harzallah, and P. K. Henri Briand. A typology of ontology-based semantic measures. In *EMOI-INTEROP'05, Proc. Open Interop Workshop on Enterprise Modelling and Ontologies for Interoperability*, Porto, Portugal, 2005.

[4] E. Blanchard, P. Kuntz, M. Harzallah, and H. Briand. A tree-based similarity for evaluating concept proximities in an ontology. In *Proc. 10th conference of the International Fedederation of Classification Societies*, pages 3–11, Ljubljana, Slovenia, 2006.

[5] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. In *Proc. 7th International Conference on World Wide Web (WWW7)*, pages 107–117, Brisbane, Australia, 1998.

[6] J. Callan and T. Mitamura. Knowledge-based extraction of named entities. In *Proc. 11th ACM Conference on Information and Knowledge Management*, pages 532–537, McLean, VA, 2002.

[7] W. W. Cohen and S. Sarawagi. Exploiting dictionaries in named entity extraction: combining semi-Markov extraction processes and data integration methods. In *Proc. 10th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 89–98, Seattle, WA, 2004.

[8] S. Cucerzan. Large-scale named entity disambiguation based on Wikipedia data. In *Proc. 2007 Joint Conference on EMNLP and CNLL*, pages 708–716, Prague, The Czech Republic, 2007.

[9] S. Cucerzan and D. Yarowsky. Language independent named entity recognition combining morphological and contextual evidence. In *Proc. 1999 Joint SIGDAT Conference on EMNLP and VLC*, pages 90–99, Maryland, MD, 1999.

[10] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. Gate: an architecture for development of robust HLT applications. In *ACL '02: Proc. 40th Annual Meeting on Association for Computational Linguistics*, pages 168–175, Philadelphia, PA, 2001.

[11] A. P. de Vries and N. Craswell. Entity ranking – guidelines. In *INEX 2006 Workshop Pre-Proceedings*, pages 413–414, 2006.

[12] A. P. de Vries, J. A. Thom, A.-M. Vercoustre, N. Craswell, and M. Lalmas. INEX 2007 Entity ranking track guidelines. In *INEX 2007 Workshop Pre-Proceedings*, 2007 (to appear).

[13] L. Denoyer and P. Gallinari. The Wikipedia XML corpus. *SIGIR Forum*, 40(1):64–69, 2006.

[14] J. Hassell, B. Aleman-Meza, and I. B. Arpinar. Ontology-driven automatic entity disambiguation in unstructured text. In *Proc. 5th International Semantic Web Conference (ISWC)*, volume 4273 of *LNCS*, pages 44–57, Athens, GA, 2006.

[15] J. M. Kleinberg. Authoritative sources in hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[16] N. Kushmerick. Wrapper induction: efficiency and expressiveness. *Artificial Intelligence*, 118(1-2):15–68, 2000.

[17] K. Lerman, S. N. Minton, and C. A. Knoblock. Wrapper maintenance: A machine learning approach. *Journal of Artificial Intelligence Research*, 18:149–181, 2003.

[18] B. Liu, R. Grossman, and Y. Zhai. Mining data records in web pages. In *Proc. 9th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–606, Washington, DC, 2003.

[19] S. Malik, A. Trotman, and M. Lalmas. Overview of INEX 2006. In *Comparative Evaluation of XML Information Retrieval Systems: 5th Workshop of the INitiative for the Evaluation of XML Retrieval, INEX 2006*, volume 4518 of *LNCS*, pages 1–11, 2007.

[20] P. McNamee and J. Mayfield. Entity extraction without language-specific resources. In *COLING-02: proceeding of the 6th conference on Natural language learning*, pages 1–4, Morristown, NJ, 2002.

[21] NIST Speech Group. The ACE 2006 evaluation plan: Evaluation of the detection and recognition of ACE entities, values, temporal expressions, relations, and events, 2006. http://www.nist.gov/speech/tests/ace/ace06/doc/ace06-evalplan.pdf.

[22] J. Pehcevski, J. A. Thom, and A.-M. Vercoustre. Hybrid XML retrieval: Combining information retrieval and a native XML database. *Information Retrieval*, 8(4):571–600, 2005.

[23] B. Popov, A. Kiryakov, D. Manov, A. Kirilov, D. Ognyanoff, and M. Goranov. Towards semantic web information extraction. In *2nd International Semantic Web Conference: Workshop on Human Language Technology for the Semantic Web and Web Services*, 2003. http://gate.ac.uk/conferences/iswc2003/proceedings/popov.pdf.

[24] A. Sahuguet and F. Azavant. Building light-weight wrappers for legacy web data-sources using W4F. In *Proc. 25th International Conference on Very Large Data Bases*, pages 738–741, Edinburgh, Scotland, UK, 1999.

[25] S. Sekine. Named entity: History and future. Technical report, Proteus Project Report, 2004. http://cs.nyu.edu/ sekine/papers/NEsurvey200402.pdf.

[26] B. Sundheim, editor. *Proc. 3rd Message Understanding Conference (MUC)*, Los Altos, CA, 1991. Morgan Kaufmann.

[27] S. Tenier, A. Napoli, X. Polanco, and Y. Toussaint. Annotation semantique de pages web. In *6mes journes francophones "Extraction et Gestion de Connaissances" - EGC 2006*, 2006.

[28] A.-M. Vercoustre and F. Paradis. A descriptive language for information object reuse through virtual documents. In *4th International Conference on Object-Oriented Information Systems (OOIS'97)*, pages 299–311, Brisbane, Australia, 1997.

[29] E. M. Voorhees and D. K. Harman. *TREC: Experiment and Evaluation in Information Retrieval*. MIT Press, 2005.

[30] J. Yu, J. A. Thom, and A. Tam. Ontology evaluation using Wikipedia categories for browsing. In *Proc. 16th ACM Conference on Information and Knowledge Management*, Lisboa, Portugal, 2007 (to appear).