Use of Wikipedia Categories in Entity Ranking

James Thom, Jovan Pehcevski, Anne-Marie Vercoustre

Use of Wikipedia Categories in Entity Ranking

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

Jovan Pehcevski
Anne-Marie Vercoustre
INRIA
Rocquencourt, France
{jovan.pehcevski, anne-marie.vercoustre}@inria.fr

Abstract *Wikipedia is a useful source of knowledge that has many applications in language processing and knowledge representation. The Wikipedia category graph can be compared with the class hierarchy in an ontology; it has some characteristics in common as well as some differences. In this paper, we present our approach for answering entity ranking queries from the Wikipedia. In particular, we explore how to make use of Wikipedia categories to improve entity ranking effectiveness. Our experiments show that using categories of example entities works significantly better than using loosely defined target categories.*

1 Introduction

Semi-structured text documents contain references to many named entities but, unlike fields in a well-structured database system, it is hard to identify the named entities within 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 (e.g. "The president of the Republic"), entity disambiguation (e.g. which Michael Jackson), question-answering, expert search, and entity ranking (also known as entity retrieval).

Entity ranking is very different from the related problem of entity extraction. The objective of *entity extraction* is to identify named entities from plain text and tag each and every occurrence; whereas the objective of *entity ranking* is to search for entities in a semi-structured collection and to get back a list of the relevant entity names as answers to a query (with possibly a page or some description associated with each entity).

The Initiative for the Evaluation of XML retrieval (INEX) is running a new track on entity ranking in 2007 [7], using Wikipedia as its document collection. In Wikipedia, pages correspond to entities which are organised into (or attached to) categories. References to entities and their categories occur frequently in natural language. For example, "France" is an named entity that corresponds to the Wikipedia page about France, belonging to categories such as "European Countries" and "Republics".

There are two tasks in the INEX 2007 entity ranking track: a task where the category of the expected entity answers is provided; and a task where a few (two or three) 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 the task of full-text retrieval, and the combination of the query and example entities (in the second task) makes it a task quite different from the task addressed by an application such as Google Sets[1] where only entity examples are provided.

In this paper, we present our approach to entity ranking that augments the initial full-text information retrieval approach with information based on hypertext links and Wikipedia categories. In our previous work we have shown the benefits of using categories in entity ranking compared to full-text retrieval [19]. Here we particularly focus on how best to use the Wikipedia category information to improve entity ranking.

2 Related Work

The traditional entity extraction problem lies in the ability to extract named entities from plain text using natural language processing techniques or statistical methods and intensive training from large collections. Benchmarks for evaluation of entity extraction have been performed for the Message Understanding Conference (MUC) [17] and for the Automatic Content Extraction (ACE) program [11].

Entity extraction

McNamee and Mayfield [14] 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 [6] 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.

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

Proceedings of the 12th Australasian Document Computing Symposium, Melbourne, Australia, December 10, 2007.

Other approaches for entity extraction are based on the use of external resources, such as an ontology or a dictionary. Popov et al. [16] use a populated ontology for entity extraction, while Cohen and Sarawagi [4] exploit a dictionary for named entity extraction. Tenier et al. [18] use an ontology for automatic semantic annotation of web pages. Their system first identifies the syntactic structure that characterises an entity in a page. It then uses subsumption to identify the more specific concept for this entity, combined with reasoning exploiting the formal structure of the ontology.

Using ontology for entity disambiguation

Hassell et al. [12] use a "populated ontology" to assist in disambiguation of entities, for example 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 structural properties of the Wikipedia.

Cucerzan [5] 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 out to 540 thousand pairs. Lexico-syntactic patterns, such as titles, links, paragraphs and lists, are used to build co-references 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.

Ontology similarity

Since Wikipedia has some but not all characteristics associated with an ontology, one could think of adapting some of the similarity measures proposed for comparing concepts in an ontology and use those for comparing categories in Wikipedia. Ehrig et al. [9] and Blanchard et al. [2] have surveyed various such similarity measures. These measures are mostly reflexive and symmetric [9] and take into account the distance (in the path) between the concepts, the depth from the root of the ontology and the common ancestor of the concepts, and the density of concepts on the paths between the concepts and from the root of the ontology [2].

All these measures rely on a strong hierarchy of the ontology concepts and a subsumption hypothesis in the parent-child relationship. Since those hypothesis are nor verified in Wikipedia (see Section 4), we could not use those similarity functions. Instead we experimented with similarities between sets of categories and lexical similarities between category names.

```
<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="185">european countries
   </category>
</categories>
</inex_topic>
```

Figure 1: Example INEX 2007 XML entity ranking topic

Entity ranking

Fissaha Adafre et al. [10] form entity neighbourhoods for every entity, which are based on clustering of similar Wikipedia pages using a combination of extracts from text content and following both incoming and outgoing page links. These entity neighbourhoods are then used as the basis for retrieval for the two entity ranking tasks.

Our approach is similar in that it uses XML structural patterns (links) rather than textual ones to identify potential entities. It also relies on the co-location of entity names with some of the entity examples (when provided). However, we also make use of the category hierarchy to better match the result entities with the expected class of the entities to retrieve.

3 INEX 2007 XML entity ranking track

The INEX XML entity ranking track is a new track that is being proposed in 2007. The track will use the Wikipedia XML document collection as its test collection.

Two tasks are planned for the INEX Entity ranking track in 2007 [7]:

task 1: *entity ranking*, which aims to retrieve entities of a given category that satisfy a topic described in natural language text; and

task 2: *list completion*, where given a topic text and a number of examples, the aim is to complete this partial list of answers.

An example of an INEX entity ranking topic is shown in Figure 1. In this example, the `title` field contains the plain content only query, the

"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."

Figure 2: Extract from the Euro Wikipedia page

`description` provides a natural language description of the information need, and the `narrative` provides a detailed explanation of what makes an entity answer relevant. In addition to these fields, 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 Wikipedia XML document collection

As Wikipedia is fast growing and evolving it is not possible to use the actual online Wikipedia for experiments, and so there is a need to use a stable collection to do evaluation experiments that can be compared over time. Denoyer and Gallinari [8] have developed an XML-based corpus based on a snapshot of the Wikipedia, which has been used by various INEX tracks in 2006 and 2007. It differs from the real Wikipedia in some respects (size, document format, category tables), but it is a very realistic approximation.

Entities in Wikipedia

In Wikipedia, an *entity* is generally associated with an article (a Wikipedia page) describing this entity. For example, there is a page for every country, most famous people or organisations, places to visit, and so forth. In Wikipedia nearly everything can be seen as an entity with an associated page. The Wikipedia is also a rich source of links, though some are to pages that do not exist yet! When mentioning entities 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.

For example, in the small extract from Euro page shown in Figure 2, there are 18 links (shown as underlined) to other pages in the Wikipedia, of which 15 are links to *countries*.

Categories in Wikipedia

Wikipedia also offers categories that authors can associate with Wikipedia pages. 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).

Wikipedia categories have unique names (e.g. "France", "European Countries"). 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).

Some properties of Wikipedia categories include:

- a category may have many subcategories 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; for example, the page Spain does not belong to the category "European countries";
- the sub-category relation is not always a subsumption relationship; for example, "Maps of Europe" is a sub-category of "Europe", but the two categories are not in an *is-a* relationship; and
- there are cycles in the category graph.

Yu et al. [20] explore these properties in more detail.

5 Category similarity approaches

To make use of the Wikipedia categories in entity ranking, we define similarity functions between:

- categories of answer entities and target categories (for task 1), and
- categories of answer entities and a set of categories attached to the entity examples (for task 2).

Task 1

We first define a very basic similarity function that computes the ratio of common categories between the set of categories $\mathsf{cat}(t)$ associated to an answer entity page t and the set $\mathsf{cat}(C)$ which is the union of the provided target categories C:

$$S_C(t) = \frac{|\mathsf{cat}(t) \cap \mathsf{cat}(C)|}{|\mathsf{cat}(C)|} \quad (1)$$

The target categories will be generally very broad, so it is to be expected that the answer entities would not generally belong to these broad categories. Accordingly, we defined several extensions of the set of categories, both for the target categories and the categories attached to answer entities.

The extensions are based on using sub-categories and parent categories in the graph of Wikipedia categories. We define $\mathsf{cat_d}(C)$ as the set containing the target category and its sub-categories (one level down) and $\mathsf{cat_u}(t)$ as the set containing the categories attached

to an answer entity t and their parent categories (one level up). Similarity function can then be defined using the same ratio as above except that $\mathsf{cat}(t)$ is replaced with $\mathsf{cat_u}(t)$ and $\mathsf{cat}(C)$ with $\mathsf{cat_d}(C)$.

Another approach is to use lexical similarity between categories. For example, "european countries" is lexically similar to "countries" since they both contain the word "countries" in their names. We use an information retrieval approach to retrieve similar categories: we have indexed all the categories using their names as corresponding documents. By sending the category names C as a query to the search engine, we then retrieve all the categories that are lexically similar to C.

We keep the top M ranked categories and add them to C to form the set $\mathsf{Ccat}(C)$. We then use the same similarity function as before, where $\mathsf{cat}(C)$ is replaced with $\mathsf{Ccat}(C)$. We also experiment with two alternative approaches: by sending the title of the topic T as a query to the search engine (denoted as $\mathsf{Tcat}(C)$); and by sending both the title of the topic T and the category names C as a query to the search engine (denoted as $\mathsf{TCcat}(C)$).

An alternative approach of using lexical similarity between categories is to index the categories using their names and the names of all their attached entities as corresponding documents. For example, if C="countries", the retrieved set of categories $\mathsf{Ccat}(C)$ may contain not only the categories that contain "countries" in their names, but also categories attached to entities with names lexically similar to "countries".

Task 2

In task 2, the categories attached to entity examples are likely to correspond to very specific categories, just like those attached to the answer entities. We define a similarity function that computes the ratio of common categories between the set of categories attached to an answer entity page $\mathsf{cat}(t)$ and the set of the union of the categories attached to entity examples $\mathsf{cat}(E)$:

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

We also expand the two sets of categories by adding the parent categories to calculate $\mathsf{cat_u}(t)$ and $\mathsf{cat_u}(E)$ and apply the same similarity function as above.

6 Our approach to entity ranking

Our approach to identifying and ranking entities combines: (1) the full-text similarity of the entity page with the query; (2) the similarity of the page's categories with the target categories or the categories of the entity examples; and (3) the links to a page from the top ranked pages returned by a search engine for the query.

6.1 Architecture

The approach involves several modules and functions that are used for processing a query, submitting it to the search engine, applying our entity ranking algorithms, and finally returning a ranked list of entities. We use Zettair[2] as our choice for a full-text search engine. Zettair is a full-text information retrieval system developed by RMIT, which returns pages ranked by their similarity score to the query. We used the Okapi BM25 similarity measure that has proved to work well on the INEX 2006 Wikipedia test collection [1].

Our approach involves the following modules:

- The search module sends the query to Zettair and returns a list of scored Wikipedia pages. The assumption is that a good entity page is a page that answers the query.
- The link extraction module extracts the links from a selected number of highly ranked pages,[3] together with the information about the paths of the links (XML paths). The assumption is that a good entity page is a page that is referred to by a page answering the query; this is an adaptation of the Google PageRank [3] and HITS [13] algorithms to the problem of entity ranking.
- The linkrank module calculates a weight for a page based (among other things) on the number of links to this page (see 6.2). The assumption is that a good entity page is a page that is referred to from contexts with many occurrences of the entity examples. A coarse context would be the full page that contains the entity examples. Smaller and better contexts may be elements such as paragraphs, lists, or tables [15].
- The category similarity module calculates a weight for a page based on the similarity of the page categories with that of the target categories or the categories attached to the entity examples (see 6.3). The assumption is that a good entity page is a page associated with a category close to the target categories or categories of the entity examples.
- The full-text retrieval module calculates a weight for a page based on its initial Zettair score (see 6.4).

The global score for a page is calculated as a linear combination of three normalised scores coming out of the last three modules (see 6.5).

The above architecture provides a general framework for evaluating entity ranking.

6.2 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

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

[3] We discarded external links and some internal collection links that do not refer to existing pages in the INEX Wikipedia collection.

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.

We use a very basic linkrank function that, for an answer entity page t that is pointed to by a page p, takes into account the Zettair score of the referring page $z(p)$, and the number of reference links to the answer entity page $\#links(p,t)$:

$$S_L(t) = \sum_{r=1}^{N} z(p_r) * f(\#links(p_r, t)) \quad (3)$$

where $f(x) = x$ (when there is no reference link to the answer entity page, $f(x) = 0$).

The linkrank function can be implemented in a variety of ways; for task 2 where entity examples are provided, we have also experimented by weighting pages containing a number of entity examples, or by exploiting the locality of links around the entity examples [15]. This more complex implementation of the linkrank function is outside the scope of this paper.

6.3 Category score

The basic category score $S_C(t)$ is calculated for the two tasks as follows:

task 1

$$S_C(t) = \frac{|\mathsf{cat}(t) \cap \mathsf{cat}(C)|}{|\mathsf{cat}(C)|} \quad (4)$$

task 2

$$S_C(t) = \frac{|\mathsf{cat}(t) \cap \mathsf{cat}(E)|}{|\mathsf{cat}(E)|} \quad (5)$$

We then consider variations on the category score $S_C(t)$ given the considerations in Section 5, using various combinations of replacing $\mathsf{cat}(t)$ with $\mathsf{cat_u}(t)$, replacing $\mathsf{cat}(C)$ with $\mathsf{cat_d}(C)$, $\mathsf{Ccat}(C)$, $\mathsf{Tcat}(C)$ or $\mathsf{TCcat}(C)$, and replacing $\mathsf{cat}(E)$ with $\mathsf{cat_u}(E)$.

6.4 Z score

The Z score assigns the initial Zettair score to an answer entity page. If the answer page does not appear in the final list of ranked pages returned by Zettair, then its Z score is zero:

$$S_Z(t) = \begin{cases} z(t) & \text{if page } t \text{ was returned by Zettair} \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

6.5 Global score

The global score $S(t)$ for an answer 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 $S_Z(t)$:

$$S(t) = \alpha S_L(t) + \beta S_C(t) + (1 - \alpha - \beta) S_Z(t) \quad (7)$$

where α and β are parameters that can be tuned.

A special case of interest here is when only the category score is used ($\alpha = 0.0$, $\beta = 1.0$). It allows us to evaluate the effectiveness of various category similarity functions and the overall benefit of using categories.

7 Experimental results

We now present results that investigate the effectiveness of our entity ranking approach for the two entity ranking tasks. We start by describing the test collection we developed for entity ranking.

7.1 Test collection

There is no existing set of topics with relevance assessments for entity ranking, although such a set will be developed for the INEX XML entity ranking track in 2007. So for these experiments we developed our own test collection based on a selection of topics from the INEX 2006 ad hoc track. We chose 27 topics that we considered were of an "entity ranking" nature, where for each page that had been assessed as containing relevant information, we reassessed whether or not it was an entity answer, and whether it *loosely* belonged to a category of entities we had *loosely* identified as being the target of the topic. If there were entity examples mentioned in the original topic 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. To this set of 27 topics we also added the Euro topic example (shown in Figure 1) that we had created by hand from the original INEX description of the entity ranking track [7], resulting in total of 28 entity ranking topics.

We use mean average precision (MAP) as our primary method of evaluation, but also report results using several alternative information retrieval measures: mean of P[5] and P[10] (mean precision at top 5 or 10 entities returned), and 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). When dealing with entity ranking, the ultimate goal is to retrieve all the answer entities at the top of the ranking. Although we believe that MAP may be more suitable than the other measures in capturing these aspects, part of the track at INEX 2007 will involve determining what is the most suitable measure.

7.2 Task 1

For this task we carried out three separate investigations. First, we wanted to investigate the effectiveness of our category similarity module when varying the extensions of the set of categories attached to both the target categories and the answer entities. We also investigated the impact that this variation had on the effectiveness when the two different category indexes are

Table 1: Performance scores for runs using different retrieval strategies in our category similarity module (α0.0–β1.0), obtained for task 1 by different evaluation measures. For the three runs using lexical similarity, the Zettair index comprises documents containing category names (C), or documents containing category names and names of entities associated with the category (CE). The number of category answers retrieved by Zettair is M=10. For each measure, the best performing score is shown in bold.

Run	P[r] 5	P[r] 10	R-prec	MAP
$\mathsf{cat}(C)$-$\mathsf{cat}(t)$	0.229	0.250	0.215	0.196
$\mathsf{cat_d}(C)$-$\mathsf{cat_u}(t)$	0.243	0.246	0.209	0.185
$\mathsf{Ccat}(C)$-$\mathsf{cat}(t)$	0.214	0.250	0.214	0.197
$\mathsf{Tcat}(C)$-$\mathsf{cat}(t)$	**0.264**	0.261	0.239	0.216
$\mathsf{TCcat}(C)$-$\mathsf{cat}(t)$	**0.264**	**0.286**	**0.247**	**0.226**

(C) Index of category names

Run	P[r] 5	P[r] 10	R-prec	MAP
$\mathsf{cat}(C)$-$\mathsf{cat}(t)$	0.229	**0.250**	**0.215**	**0.196**
$\mathsf{cat_d}(C)$-$\mathsf{cat_u}(t)$	**0.243**	0.246	0.209	0.185
$\mathsf{Ccat}(C)$-$\mathsf{cat}(t)$	0.157	0.171	0.149	0.148
$\mathsf{Tcat}(C)$-$\mathsf{cat}(t)$	0.171	0.182	0.170	0.157
$\mathsf{TCcat}(C)$-$\mathsf{cat}(t)$	0.207	0.214	0.175	0.173

(CE) Index of category and entity names

used by Zettair. Second, for the best category similarity approach we investigated the optimal value for the parameter M (the number of category answers retrieved by Zettair). Last, for the best category similarity approach using the optimal M value, we investigated the optimal values for the α and β parameters. The aim of this investigation is to find the best score that could be achieved by our entity ranking approach for task 1.

Investigating category similarity approaches

The results of these investigations are shown in Tables 1(C) and 1(CE).[4] Several observations can be drawn from these results.

First, the choice of using the Zettair category index can dramatically influence the entity ranking performance. When cross-comparing the results in the two tables, we observe that the three lexical similarity runs using the Zettair index of category names substantially outperform the corresponding runs using the Zettair index of category and entity names. The differences in performance are all statistically significant ($p < 0.05$). Second, the run that uses the query that combines the terms from the title and the category fields of an INEX topic ($\mathsf{TCcat}(C)$-$\mathsf{cat}(t)$) performs the best among the three runs using lexical similarity, and overall it also performs the best among the five runs when using the Zettair index of category names. However, the differences in performance between this and the other four runs are not statistically significant. Third, extending the set of categories attached to both the target categories and the answer entities overall does not result in an improved performance, although there are some (non-significant) early precision improvements.

Investigating the parameter M

The above results show that the best effectiveness for our category similarity module (α0.0–β1.0) is achieved when using the Zettair index of category names, together with the query strategy that combines the terms from the title and the category fields of an INEX topic.

[4]The first two runs do not use any of Zettair's category indexes and are included for comparison.

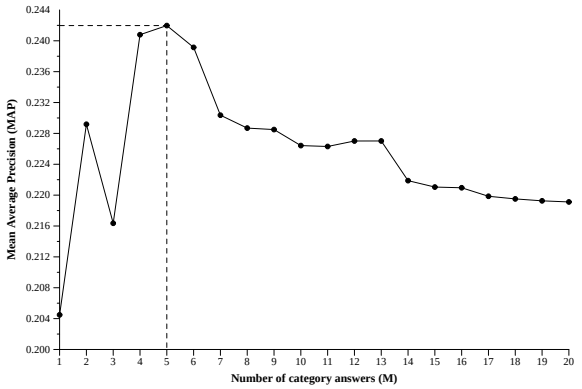

Figure 3: Investigating the optimal value for the number of category answers retrieved by Zettair, when using the run $\mathsf{TCcat}(C)$-$\mathsf{cat}(t)$.

For these experiments we used a fixed value M=10 for the parameter M that represents the number of category answers retrieved by Zettair. However, since this was an arbitrary choice we also investigated whether a different value of M could also have a positive impact on the retrieval effectiveness. We therefore varied M from 1 to 20 in steps of 1, and measured the MAP scores achieved by our best performing $\mathsf{TCcat}(C)$-$\mathsf{cat}(t)$ run using the Zettair index of category names.

Figure 3 shows the results of this investigation. We observe that a value of 5 for the parameter M yields the highest MAP score (0.242) for our category similarity module, which is a 7% relative performance improvement over the MAP score obtained with M=10. This performance improvement is statistically significant ($p < 0.05$).

Investigating the combining parameters α and β

To find the best score that could be achieved by our entity ranking approach for task 1, we used the run $\mathsf{TCcat}(C)$-$\mathsf{cat}(t)$ with the optimal value M=5 and investigated various combinations of scores obtained from the three modules. We calculated MAP over the 28 topics in our test collection, as we varied α from 0 to 1 in steps of 0.1. For each value of α, we also varied β from 0 to $(1 - \alpha)$ in steps of 0.1. We

Table 2: Performance scores for runs using different retrieval strategies in our category similarity module (α0.0–β1.0), obtained for task 2 by different evaluation measures. For each measure, the best performing score is shown in bold.

Run	P[r] 5	P[r] 10	R-prec	MAP
$\mathsf{cat}(E)$-$\mathsf{cat}(t)$	**0.536**	**0.393**	**0.332**	**0.338**
$\mathsf{cat}(E)$-$\mathsf{cat_u}(t)$	0.493	0.361	0.294	0.313
$\mathsf{cat_u}(E)$-$\mathsf{cat}(t)$	0.407	0.336	0.275	0.255
$\mathsf{cat_u}(E)$-$\mathsf{cat_u}(t)$	0.357	0.332	0.269	0.261

found that the highest MAP score (0.287) is achieved for $\alpha = 0.1$ and $\beta = 0.8$. This is a 19% relative performance improvement over the best score achieved by using only the category module (α0.0–β1.0). This performance improvement is statistically significant ($p < 0.05$). We also calculated the scores using mean R-precision instead of MAP as our evaluation measure, and we again observed the same performance behaviour and optimal values for the two parameters.

7.3 Task 2

For this task we carried out two separate investigations. First, as with task 1 we wanted to investigate the effectiveness of our category similarity module when varying the extensions of the set of categories attached to both the example and the answer entities. Second, for the best category similarity approach we investigated the optimal values for the α and β parameters, with the aim of finding the best score that could be achieved by our entity ranking approach for task 2.

Investigating category similarity approaches

The results of these investigations are shown in Table 2. We observe that, as with task 1, extending the set of categories attached to either (or both) of the example and answer entities does not result in an improved performance. The differences in performance between the best performing run that does not use the extended category sets and the other three runs that use any (or both) of these sets are all statistically significant ($p < 0.05$).

Investigating the combining parameters α and β

To find the best score that could be achieved by our entity ranking approach for task 2, we used the run $\mathsf{cat}(E)$-$\mathsf{cat}(t)$ and investigated various combinations of scores obtained from the three modules. We calculated MAP over the 28 topics in our test collection, as we used the 66 combined values for parameters α and β. We found that the highest MAP score (0.396) was again achieved for $\alpha = 0.1$ and $\beta = 0.8$. This score is a 17% relative performance improvement over the best score achieved by using only the category module (α0.0–β1.0). The performance improvement is statistically significant ($p < 0.05$).

Table 3: Comparing best performing runs for task 1 and task 2 for two distinct cases (using either category or global scores). The number of category answers retrieved by Zettair for run $\mathsf{TCcat}(C)$-$\mathsf{cat}(t)$ is M=5. For each case, the best results are shown in bold.

Run	P[r] 5	P[r] 10	R-prec	MAP
Category score: α0.0–β1.0				
$\mathsf{TCcat}(C)$-$\mathsf{cat}(t)$	0.307	0.318	0.263	0.242
$\mathsf{cat}(E)$-$\mathsf{cat}(t)$	**0.536**	**0.393**	**0.332**	**0.338**
Global score: α0.1–β0.8				
$\mathsf{TCcat}(C)$-$\mathsf{cat}(t)$	0.379	0.361	0.338	0.287
$\mathsf{cat}(E)$-$\mathsf{cat}(t)$	**0.607**	**0.457**	**0.412**	**0.396**

7.4 Comparing Task 1 and Task 2

To investigate which of the two query strategies (target categories or example entities) is more effective for entity ranking, we compared the scores of the best performing runs across the two tasks. Table 3 shows the results of this comparison, when separately taking into account two distinct cases: a case when using scores coming out of the category module only (α0.0–β1.0); and a case when using optimal global scores coming out of the three modules (α0.1–β0.8).

We observe that, irrespective of whether category or global scores are used by our entity ranking approach, the run that uses the set of categories attached to example entities (task 2) substantially outperforms the run that uses the set of categories identified by Zettair using the topic title and the target categories (task 1). The differences in performance between the two runs are statistically significant ($p < 0.05$). This finding shows that using example entities is much more effective query strategy than using the loosely defined target categories, which allows for the answer entities to be identified and ranked more accurately.

8 Conclusions and future work

In this paper, we have presented our entity ranking approach for the INEX Wikipedia XML document collection. We focused on different entity ranking strategies that can be used by our category similarity module, and evaluated these strategies on the two entity ranking tasks. For task 1, we demonstrated that using lexical similarity between category names results in an effective entity ranking approach, so long as the category index comprises documents containing only category names. For task 2, we demonstrated that the best approach is the one that uses the sets of categories directly attached to both the example and the answer entities, and that using various extensions of these two sets significantly decreases the entity ranking performance. For the two tasks, combining scores coming out of the three modules significantly improves the performance

compared to that achieved when using scores only from the category module. Importantly, when comparing the scores of the best performing runs across the two tasks, we found that the query strategy that uses example entities to identify the set of target categories is significantly more effective than the strategy that uses the set of loosely defined target categories.

In the future, we plan to further improve the global score of our entity ranking approach by using a better linkrank function that exploits different (static and dynamic) contexts identified around the entity examples. Preliminary results demonstrate that the locality of links around entity examples can indeed be exploited to significantly improve the entity ranking performance compared to the performance achieved when using the full page context [15]. To improve the category similarity function, we plan to introduce different category weighting rules that we hope would better distinguish the answer entities that are more similar to the entity examples. 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.

Acknowledgements Part of this work was completed while James Thom was visiting INRIA in 2007.

References

[1] D.N.F. Awang Iskandar, Jovan Pehcevski, James A. Thom and S. M. M. Tahaghoghi. Social media retrieval using image features and structured text. In *Comparative Evaluation of XML Information Retrieval Systems: Fifth Workshop of the INitiative for the Evaluation of XML Retrieval, INEX 2006*, Volume 4518 of *Lecture Notes in Computer Science*, pages 358–372, 2007.

[2] Emmanuel Blanchard, Mounira Harzallah and Pascale Kuntz Henri Briand. A typology of ontology-based semantic measures. In *EMOI-INTEROP'05, Proceedings of the Open Interop Workshop on Enterprise Modelling and Ontologies for Interoperability*, Porto, Portugal, 2005.

[3] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual Web search engine. In *Proceedings of the 7th International Conference on World Wide Web*, pages 107–117, Brisbane, Australia, 1998.

[4] William W. Cohen and Sunita Sarawagi. Exploiting dictionaries in named entity extraction: combining semi-Markov extraction processes and data integration methods. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 89–98, Seattle, WA, USA, 2004.

[5] Silviu Cucerzan. Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the 2007 Joint Conference on EMNLP and CoNLL*, pages 708–716, Prague, Czech Republic, 2007.

[6] Silviu Cucerzan and David Yarowsky. Language independent named entity recognition combining morphological and contextual evidence. In *Proceedings of the 1999 Joint SIGDAT Conference on EMNLP and VLC*, pages 90–99, Maryland, MD, USA, 1999.

[7] Arjen P. de Vries, James A. Thom, Anne-Marie Vercoustre, Nick Craswell and Mounia Lalmas. INEX 2007 Entity ranking track guidelines. In *INEX 2007 Workshop Pre-Proceedings*, 2007 (to appear).

[8] Ludovic Denoyer and Patrick Gallinari. The Wikipedia XML corpus. *SIGIR Forum*, Volume 40, Number 1, pages 64–69, 2006.

[9] Marc Ehrig, Peter Haase, Nenad Stojanovic and Mark Hefke. Similarity for ontologies — a comprehensive framework. In *Proceedings of the 13th European Conference on Information Systems*, 2005.

[10] Sisay Fissaha Adafre, Maarten de Rijke and Erik Tjong Kim Sang. Entity retrieval. In *Proceedings of International Conference on Recent Advances in Natural Language Processing (RANLP - 2007), September 27-29, Borovets, Bulgaria*, 2007.

[11] NIST Speech Group. The ACE 2006 evaluation plan: Evaluation of the detection and recognition of ACE entities, values, temporal expressions, relations, and events, 2006.

[12] Joseph Hassell, Boanerges Aleman-Meza and I. Budak Arpinar. Ontology-driven automatic entity disambiguation in unstructured text. In *Proceedings of the 5th International Semantic Web Conference (ISWC)*, Volume 4273 of *Lecture Notes in Computer Science*, pages 44–57, Athens, GA, USA, 2006.

[13] Jon M. Kleinberg. Authoritative sources in hyperlinked environment. *Journal of the ACM*, Volume 46, Number 5, pages 604–632, 1999.

[14] Paul McNamee and James Mayfield. Entity extraction without language-specific resources. In *COLING-02: Proceedings of the 6th Conference on Natural Language Learning*, pages 1–4, Morristown, NJ, USA, 2002.

[15] Jovan Pehcevski, Anne-Marie Vercoustre and James A. Thom. Exploiting locality of Wikipedia links in entity ranking. Submitted for publication, 2007.

[16] Borislav Popov, Atanas Kiryakov, Dimitar Manov, Angel Kirilov, Damyan Ognyanoff and Miroslav Goranov. Towards semantic web information extraction. In *2nd International Semantic Web Conference: Workshop on Human Language Technology for the Semantic Web and Web Services*, pages 1–22, Florida, USA, 2003.

[17] B. Sundheim (editor). *Proceedings of the Third Message Understanding Conference (MUC)*, Los Altos, CA, 1991. Morgan Kaufmann.

[18] Sylvain Tenier, Amedeo Napoli, Xavier Polanco and Yannick Toussaint. Annotation semantique de pages web. In *6emes journees francophones Extraction et Gestion de Connaissances (EGC)*, Lille, France, 2006.

[19] Anne-Marie Vercoustre, James A. Thom and Jovan Pehcevski. Entity ranking in Wikipedia. In *Proceedings of the 23rd Annual ACM Symposium on Applied Computing (SAC08)*, Fortaleza, Brazil, 2008 (to appear).

[20] Jonathan Yu, James A. Thom and Audrey Tam. Ontology evaluation using Wikipedia categories for browsing. In *Proceedings of 16th ACM Conference on Information and Knowledge Management (CIKM 2007)*, pages 223–232, Lisboa, Portugal, 2007.