HAL open science

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. [Research Report] RR-6294, 2007, pp.8. inria-00172511v1

HAL Id: inria-00172511

https://inria.hal.science/inria-00172511v1

Submitted on 17 Sep 2007 (v1), last revised 18 Sep 2007 (v2)

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.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Entity Ranking in Wikipedia

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

N° ????

Septembre 2007

Thème COG

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--????--FR+ENG

Entity Ranking in Wikipedia

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

Thème COG — Systèmes cognitifs
Projet AxIS

Rapport de recherche n° ???? — Septembre 2007 — 8 pages

Abstract: The traditional entity extraction problem is to extract 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 for 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. The paper also introduces our methodology for evaluating the effectiveness of entity ranking, as well as preliminary results which show that the use of categories and the link structure of Wikipedia, together with entity examples, can significantly improve retrieval effectiveness.

Key-words: Entity Ranking, Test collection, XML Retrieval, Wikipedia

* INRIA, Rocquencourt, France
† RMIT, Melbourne, Australia. This work was undertaken while James Thom was invited at INRIA in 2007

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex (France)
Téléphone : +33 1 39 63 55 11 — Télécopie : +33 1 39 63 53 30

Recherche d'entités dans Wikipedia

Résumé : L'extraction d'entités nommées dans des documents textuels utilise généralement des techniques d'analyse de langue naturelle, avec apprentissage sur de grande collections de documents. Les entités nommées peuvent tre, par exemple, des noms de personnes, d'organismes, de lieux, ou des dates. L'extraction de telles entités est un domaine de recherche très actif. Nous sommes intéressés la recherche d'entité dans le domaine de la recherche d'information (IR). Dans cet article nous présentons notre approche pour identifier et classer des entités trouvées dans la collection Wikipedia utilisée par le groupe international INEX. Wikipedia offre un certain nombre de caractéristiques intéressantes pour la recherche d'entités que nous introduisons d'abord. Puis nous décrivons les principes et l'architecture de notre système de recherche d'entités. Nous présentons également la méthode utilisée pour évaluer le système, dans une tâche où les utilisateurs fournissent deux ou trois exemples de ce qu'ils recherchent. Les résultats préliminaires montrent que l'utilisation des catégories et de la structure des liens de Wikipedia, ainsi que les exemples fournis, améliorent de faon significative la qualité de la recherche (précision et rappel).

Mots-clés : Entités nommées, Large collections, XML, Wikipedia, Recherche d'information

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

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. The paper also introduces our methodology for evaluating the effectiveness of entity ranking, as well as preliminary results which 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; H.3.7 Digital Libraries

General Terms: Measurement, Performance, Experimentation

Keywords: Entity Ranking, Test collection, XML Retrieval

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 (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).

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. Benchmarks for evaluation of entity extraction have been performed for the Message Understanding Conference (MUC) [27] and for the Automatic Content Extraction (ACE) program [22]. In that context, 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. However, 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, and possibly a page or some description associated with each entity.

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 pages about the Euro and similar currencies. The Initiative for the Evaluation of XML retrieval (INEX) has proposed a new track on entity ranking [12], using Wikipedia as its document collection. Two tasks are proposed for 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 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.

This paper is organised as follows. In the following section, we review related work on wrappers, entity extraction and disambiguation, and link ranking. Sections 3 and 4 introduce the INEX entity ranking track and the INEX Wikipedia collection, respectively. In Section 5, we describe the principles and the architecture of our entity ranking system. Experimental results using training and test data sets are presented in Section 6. We then conclude the paper and outline some future work directions.

2. RELATED WORK

Our entity ranking approach gets its inspiration from wrapping technology, entity extraction, the use of ontologies for entity ex-

*Work undertaken while James Thom visited INRIA in 2007.

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

Copyright of this article remains with the authors.

traction or entity disambiguation, and information retrieval.

Wrappers

A wrapper is a tool that extracts information 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, 17, 25, 29]. 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 building experimental data sets.

Web wrappers are based on two main approaches: scripting languages [25, 29] or wrapper induction [1, 17].

Wrapper scripting languages are very close to current XML query languages: they allow for selecting information using XML paths, or more advanced queries, in semi-structured documents (XML-like) and reconstructing results into a new XML document. Such wrappers are easy to write for sets of regular pages, but would break easily even on small changes in the structure of the pages.

Inductive wrappers are built using examples from which the wrapper learns the rules for extracting the information and maps it into the appropriate data. Those wrappers need to also be retrained if the pages change.

To prevent wrappers breaking over time without notice, Lerman et al. [18] 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.

The system we propose for entity ranking also works dynamically, at query time instead of at wrapping time. We are also using weighting algorithms based on a couple of lexico-syntactic patterns (links and list-like contexts) that are well represented in web-based collections, as well as the knowledge of categories that is a specific Wikipedia feature.

Entity extraction

The main goal of entity extraction is to identify entity occurrences in plain text and to mark them with their appropriate type [22, 26]. The entity type could be, for example, a person, organisation, location, or date. There are mainly two approaches to entity extraction. The first one is grammar-based, and it uses natural language processing to identify and classify entities [10]. This approach is efficient but requires many rules (as many as 400 in some domains) handwritten by experts. The second one is statistical model-based which is more generic and flexible, but requires large collections for training.

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 [21] 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. [24] use a populated ontology for entity extraction, while Cohen and Sarawagi [7] exploit a dictionary for named entity extraction. Tenier et al. [28] 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. [15] 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 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 co-references of entities in limited contexts. However, the overwhelming number of contexts that could be extracted this way requires the use of heuristics to limit the context extraction. The knowledge extracted from Wikipedia is then used for improving entity disambiguation in the context of web and news search.

Information retrieval (PageRank and HITS)

In information retrieval (IR), the similarity of a document to a query indicates how closely the content of the document matches that of the query. To calculate the query-document similarity, most IR systems use statistical information concerning the distribution of the query terms, both within the document and the collection as a whole. However, when dealing with the World Wide Web (or other hyperlinked environments, such as Wikipedia), hyperlinks are 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 [16]. 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 also 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 also makes our link ranking algorithm to be query-dependent (just like HITS), which allows for it to be dynamically calculated at query time.

```
<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

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 [20]. Two tasks are planned for the INEX Entity ranking track in 2007 [12]:

task1: entity ranking, which aims at retrieving entities of a given category that satisfy a topic described in natural language text; and

task2: 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 2007 entity ranking topic is shown in Figure 1. In this example, the `title` field contains the plain content only query, the `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 FEATURES

In this section, we analyse several interesting Wikipedia features that may be used for entity identification and ranking. We start with a description of the INEX Wikipedia XML document collection.

4.1 The INEX Wikipedia collection

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, and so we need a stable collection to do evaluation experiments that can be compared over time. Denoyer and Gallinari [13] have developed an XML-based corpus founded on a snapshot of the Wikipedia, which has been used by various INEX 2006 tracks. It differs from the real Wikipedia in some respects (size, document format, category tables), but it is a very realistic approximation. More specifically, the INEX Wikipedia XML document collection retains the main characteristics of the online version, although they have been implemented through XML tags instead of the initial HTML tags and the native Wikipedia structure.

Table 1: INEX Wikipedia English document collection

number of articles	659,388
collection size	~ 4.6 Gigabytes
number of categories	113,483
average categories per article	2.2849

"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 page

The INEX Wikipedia XML corpus is composed of 8 main collections, corresponding to 8 different languages: English, French, German, Dutch, Spanish, Chinese, Arabian and Japanese. The INEX 2007 Entity ranking track will use the English sub-collection. Some properties of this sub-collection are shown in Table 1.

We now describe the high level features that are relevant to the problem of entity ranking, namely the notion of entity in Wikipedia, and the Wikipedia categories.

4.2 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. Some of the types of entities in Wikipedia include: Art museums and galleries, Countries, Famous people (actors, writers, explorers, etc.), Monarchs of the British Isles, Magicians, Diseases, Movies, Songs, and Books.

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 every occurrence of the entity name to the page describing this entity.[2] 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 Euro page (see Figure 2), 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 figure, there are 18 entity references of which 15 are country names; more specifically, these countries are all "European Union member states", which brings us to the notion of category in Wikipedia.

4.3 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 ad-

[2] At least it is recommended to link the first occurrence of the entity in the article. For long articles, multiple occurrences of the same entity may or may not be linked to the corresponding entity page.

ministrative categories).

As seen in Table 1, 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 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.

Yu et al. [31] explore these properties in more detail.

When searching for entities it is natural to take advantage of the Wikipedia categories since they would give a hint on whether the retrieved entities are of the expected type. For example, if you are looking for entities "authors", pages associated with the category "Novelist" may be more relevant than pages associated with the category "Book".

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). However, our approach can also be used for entity ranking where the category of the target entities is given and no examples are provided (task1).

We developed an algorithm for identifying and ranking potential entity pages that combines: (1) the full-text similarity of the entity page with the query, (2) the number of links to the entity page from the top ranked pages returned by a search engine for the query, and (3) the similarity of the page's categories with the categories of the entity examples.

5.1 Principles

Our approach is based on the following principles:

- a good entity page is a page that answers the query (or a query extended with the examples).
- a good entity page is a page associated with a category close to the categories of the entity examples. We introduce a similarity function between the categories of a page and the categories of the given examples.
- a good entity page is a page that is pointed to by a page answering the query; this is an adaptation of the HITS [16] algorithm to the problem of entity ranking. We refer to it as a linkrank algorithm.
- a good entity page is a page that is pointed to by 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 [14, 19]. In this work, we use the coarse context (the full page) when calculating the scores in our linkrank algorithm.

We have built a system based on the above principles, and a framework to tune and evaluate a set of different algorithms. More specifically, candidate pages are ranked by combining three different scores: a linkrank score, a category score, and the initial search engine similarity score.

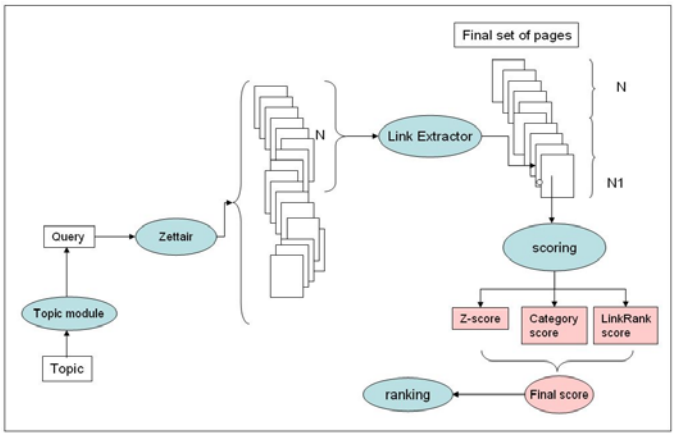

Figure 3: Process for Entity ranking

5.2 Architecture

The system 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[3] as our choice for a full-text search engine. Zettair is a full-text IR system developed by RMIT University, 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 [2].

Our system involves the following modules and functions:

- 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);
- the link extraction module extracts the links from a selected number of highly ranked pages,[4] together with the information concerning the paths of the links (XML paths);
- the linkrank module calculates a weight for a page based (among other things) on the number of links to this page (see 5.3.1);
- the category similarity module calculates a weight for a page based on the similarity of the page categories with those of the entity examples (see 5.3.2); and
- the full-text IR module calculates a weight for a page based on its initial Zettair score (see 5.3.3).

The global score for a page is calculated as a linear combination of three normalised scores coming out of the last three modules (see 5.3.4).

The overall process for entity ranking is shown in Figure 3. 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.

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

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

5.3 Score Functions and parameters

The core of our algorithm is based on combining different scoring functions for a result page, which we now describe in more detail.

5.3.1 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. The number N has been kept to a relatively small value mainly for performance purposes, since Wikipedia pages contain many links that would otherwise need to be extracted. 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 reference links to the target page $\#links(p_r, t)$, and the number of distinct entity examples in the referring page $\#ent(p_r)$:

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

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).

5.3.2 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 not necessarily a distance between two categories, but between sets of categories: the set of categories associated to a given page, and the set of categories associated to the entity examples. We use a very basic similarity function that calculates the ratio of common categories between the set of categories associated to the target page $\mathsf{cat}(t)$ and the set of the union of the categories associated to the entity examples $\mathsf{cat}(q)$:

$$S_C(t) = \frac{|\mathsf{cat}(t) \cap \mathsf{cat}(q)|}{|\mathsf{cat}(q)|} \qquad (2)$$

5.3.3 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:

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

5.3.4 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 $S_Z(t)$:

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

where α and β are two parameters that can be tuned differently depending on the entity retrieval task.

We consider some special cases that allow us to evaluate the effectiveness of each module in our system:

- $\alpha = 1, \beta = 0$ where only the linkrank score is used;
- $\alpha = 0, \beta = 1$ where only the category score is used; and
- $\alpha = 0, \beta = 0$ where only the Z score is used.[5]

More combinations of the two parameters will be explored in the training phase of our system, which is presented next.

6. EXPERIMENTAL RESULTS

In this section, we present experimental results for our entity ranking system using training and test data sets.

6.1 Methodology

We start with the methodology used in the evaluation, by describing the entity topics with their corresponding relevance assessments, and the evaluation measures used to evaluate the effectiveness of our entity ranking system.

6.1.1 Entity topics and relevance assessments

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 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 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.

These 27 topics were divided into two sets. The first 9 topics were used as our training data set, to which we added two more topics that we had created by hand from the original INEX description of the entity ranking track (one of these two extra topics is the Euro example shown in Figure 1). The remaining 18 topics were used as our test data set.

6.1.2 Evaluation measures

Evaluation measures are used to compare the retrieval performance of different systems. In this paper we use MAP (mean average precision) as our primary method of evaluation, but also report

[5]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, which are not necessarily identical to those initially returned by Zettair.

Table 2: 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*										

some results with other measures which are typically used to evaluate the retrieval performance of IR systems [30].

We calculate average precision for each topic at natural recall levels by first removing the entity examples both from the list of answers returned by each system and from the relevance assessments. We do this because the task is to find entities other than the examples provided in the topic.

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 ith 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), and mean interpolated precision at various recall levels.

6.2 Training data set (11 topics)

We used the training data set to determine suitable values for the parameters α and β, and also to try out some minor variations to our system (such as whether or not to include the names of the example entities in the query sent to Zettair).

We evaluated MAP over the 11 topics in the training set, 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. Table 2 shows these results. We observe that the highest MAP (0.2911) on the training data set is achieved for $\alpha = 0.3$ and $\beta = 0.6$ (shown in bold).

We also trained using mean R-precision instead of MAP as our evaluation measure (see Table 3), where we observed somewhat different optimal values for the two parameters: $\alpha = 0.4$ and $\beta = 0.3$. One reason for this is the relatively small number of topics used in the training data set. The optimal parameter values obtained by MAP and R-precision should converge if much larger number of training topics is used.

On the training data set, we also experimented with adding the names of the example entities to the query sent to Zettair. This generally performed worse, both for the plain Zettair system and for the systems where we extracted links and used various combinations of S_Z, S_C, and S_L scores. We plan to perform a more detailed per-topic analysis in order to investigate this (somewhat peculiar) retrieval behaviour.

6.3 Test data set (18 topics)

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

- full-text retrieval using Zettair
- 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 the global score ($0.3 * S_L + 0.6 * S_C + 0.1 * S_Z$)

The results for these five runs are shown in Table 4 and Figure 4. We observe that the two best entity ranking approaches are those that place most of the weight on the category score S_C (runs α0.0–β1.0 and α0.3–β0.6). However, of the two best runs, with both MAP and R-precision only the α0.3–β0.6 run performs significantly better ($p < 0.05$) than both the plain Zettair full-text retrieval run and the run that only uses the Z score (α0.0–β0.0) in the re-ranking. All the four runs, in turn, perform significantly better ($p < 0.05$) than the worst performing run (α1.0–β0.0) which only uses 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.

6.4 Unjudged entities

Tables 2 to 4 show that the absolute performance scores of our entity ranking runs are somewhat low. We therefore investigated whether this could be due to the properties of our training and test data sets, since we suspected that there could be a large number of unjudged entities retrieved by our system.

Table 5 shows the percentage of unjudged entities retrieved by the five runs, both among the first R retrieved entities (***R***) and

Table 3: Mean R-precision scores for runs using 66 possible α–β combinations, obtained on the 11 training topics. Queries sent to Zettair include only terms from the topic title (Q). The mean R-precision score of the plain Zettair run is 0.1596. The numbers in italics show the scores obtained for each of the three individual modules. The best performing mean R-precision score is 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	*0.1569*	0.1716	0.1743	0.1970	0.2280	0.2363	0.2296	0.2267	0.2320	0.2308	*0.2302*
0.1	0.1569	0.1708	0.1817	0.2264	0.2573	0.2569	0.2887	0.2737	0.2834	0.2372	
0.2	0.1850	0.1989	0.2127	0.2541	0.2817	0.2630	0.2838	0.2850	0.2454		
0.3	0.1975	0.2274	0.2218	0.2606	0.2911	0.2913	0.2883	0.2628			
0.4	0.2191	0.2242	0.2758	**0.3058**	0.2785	0.2952	0.2573				
0.5	0.2410	0.2532	0.2914	0.2975	0.2730	0.2559					
0.6	0.2371	0.2524	0.2706	0.2770	0.2629						
0.7	0.2042	0.2398	0.2417	0.2546							
0.8	0.1853	0.2054	0.2074								
0.9	0.1827	0.1908									
1.0	*0.1640*										

Table 4: Performance scores for runs using different evaluation measures, obtained on the 18 INEX 2006 test topics. Queries sent to Zettair include only terms from the topic title (Q). For each measure, the best performing score is 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

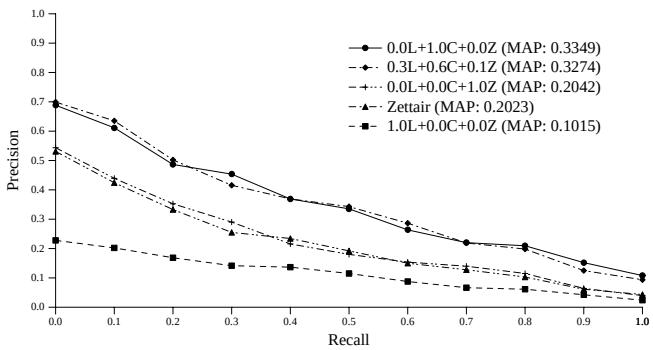

Figure 4: Evaluation of the overall performance of five runs, obtained on the 18 INEX 2006 test topics. The graph shows values for interpolated precision at 11pt recall.

Table 5: Percentage of unjudged among the first R retrieved entities (*R*) and among the total number of retrieved entities (*Total*), obtained separately on the 11 INEX 2006 training and on the 18 INEX 2006 test topics.

	Training		*Test*	
Run	***R* (%)**	***Total* (%)**	***R* (%)**	***Total* (%)**
Zettair	18	86	2	82
α0.0–β0.0	19	89	3	86
α0.0–β1.0	49	89	49	86
α1.0–β0.0	58	89	50	86
α0.3–β0.6	43	89	29	86

among the total number of retrieved entities (***Total***). The numbers were obtained separately on the 11 training and on the 18 test topics. We observe that in both cases there are relatively large numbers of unjudged among the first R entities retrieved by our entity ranking runs (on average 42% for the training and 33% for the test data set), which are between two and 16 times higher than the corresponding numbers for Zettair. These unjudged entities are assumed to be non-relevant by the evaluation measures, which in part could explain the somewhat low performance scores.

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 the use of the categories and the link structure of Wikipedia, together with the entity examples from the topic, significantly improves 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 (these could be informal lists within a paragraph or more structured lists in a list or table element). The narrower context could be defined either by fixed XML elements (such as a paragraph, a list, or a table) or it could be determined dynamically. To determine it dynamically, we plan to

identify *coherent retrieval elements* by adapting an earlier work by Pehcevski et al. [23] 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,[6] which we expect will enable us to test our approach using a larger, and more importantly, reusable set of entity ranking topics.

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: 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.

[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] T. Despeyroux, E. Fraschini, and A.-M. Vercoustre. Extraction d'entits dans des collections volutives. In *7imes Journes francophones Extraction et Gestion des Connaissances (EGC 2007), Revue des Nouvelles Technologies de l'Information (RNTI-E-3)*, pages 533–538, Namur, Belgique, 2007.

[15] 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 *Lecture Notes in Computer Science*, pages 44–57, Athens, GA, 2006.

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

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

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

[19] 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.

[20] S. Malik, A. Trotman, and M. Lalmas. Overview of INEX 2006. 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 1–11, 2007.

[21] 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.

[22] 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.

[23] 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.

[24] 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.

[25] 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.

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

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

[28] 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.

[29] 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.

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

[31] 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).

[6] http://inex.is.informatik.uni-duisburg.de/2007/xmlSearch.html

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399