Using Wikipedia Categories for Ad Hoc Search

Rianne Kaptein Marijn Koolen Jaap Kamps
University of Amsterdam, The Netherlands
kaptein@uva.nl, m.h.a.koolen@uva.nl, kamps@uva.nl

ABSTRACT

In this paper we explore the use of category information for ad hoc retrieval in Wikipedia. We show that techniques for entity ranking exploiting this category information can also be applied to ad hoc topics and lead to significant improvements. Automatically assigned target categories are good surrogates for manually assigned categories, which perform only slightly better.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Retrieval Models

General Terms

Experimentation, Measurement, Performance

Keywords

Ad hoc retrieval, Category information, Wikipedia

1. INTRODUCTION

When retrieving information from Wikipedia, we can take advantage of the specific structure of this resource. The document structure, links and categories all provide additional information that can be exploited. In this paper we will focus on category information. Category information has proved to be of great value for entity ranking in Wikipedia [5]. Improvements occur not only when manually assigned target categories are used, but also when target categories are deducted from example entities. The main difference with ad hoc retrieval is that, in entity ranking, pages can only be relevant if they are of the right entity type, whereas ad hoc retrieval places no categorical restrictions on relevance. However, if a Wikipedia category is relevant to the query topic, the pages that fall under this category may be more likely to be relevant. Our first research question is therefore:

- Can we exploit category information for ad hoc retrieval?

Since usually ad hoc topics have no target categories assigned to them, and providing target categories for entity ranking is an extra burden for users, we also examine ways to assign target categories to queries. Our second research question is:

- Can we automatically assign target categories to query topics?

In the rest of this paper we first describe our models for using category information in section 2, then our experiments in section 3 and finally our conclusion in section 4.

SIGIR'09, July 19–23, 2009, Boston, Massachusetts, USA.
ACM 978-1-60558-483-6/09/07.

2. USING CATEGORY INFORMATION

2.1 Category Assignment

To use Wikipedia's category information, we associate query topics to categories, and experiment with both manual and automatic assignment of categories. Manual assignment was done by one of the authors. The advantage of automatically assigning target categories is that it requires no user effort. There are many ways to automatically categorize topics, for example by using text categorization techniques [3]. For this paper we keep it simple and exploit the existing Wikipedia categorization of documents. From our baseline run we take the top N results, and look at the T most frequently occurring categories belonging to these documents, while requiring categories to occur at least twice. These categories are assigned as target categories to the query topic.

2.2 Retrieval Model

We use the Wikipedia categories by defining similarity functions between the categories of retrieved pages and the target categories. Pages with categories similar or equal to the target categories get a high category score. Unlike the approach in [5], where scores are estimated using lexical similarity of category names, we use similarity of pages associated with the category.

We use a language modeling approach [1] to calculate distances between categories. First of all we make a maximum likelihood estimation of the probability of a term occurring in the concatenated text of all pages belonging to that category. To account for data sparsity, we smooth the probabilities of a term occurring in a category with the background collection, which is the entire Wikipedia. The final $P(t|C)$ is estimated with a parsimonious model [2] that uses an iterative EM algorithm as follows:

$$\text{E-step:} \quad e_t = tf_{t,C} \cdot \frac{\alpha P(t|C)}{\alpha P(t|C) + (1-\alpha) P(t|D)}$$

$$\text{M-step:} \quad P(t|C) = \frac{e_t}{\sum_t e_t}, \text{i.e. normalize the model}$$

Table 1: Overlap with categories of relevant documents

N	T=1	T=2	T= 3	T=4	T=5
10	37.14	33.57	32.86	32.86	31.23
20	40.00	40.00	40.95	41.43	39.26
50	40.00	42.86	38.10	40.36	41.83
100	41.43	40.00	38.57	41.43	41.55
200	35.71	36.43	37.62	37.50	37.82

The maximum likelihood estimation of $P(t|C)$ is used as initial probability. Now we can calculate distances between categories. We do this using KL-divergence to calculate a category score that is high when the distance between the categories is small. We sum the scores of each target category, using only the minimal distance from the document categories to a target category. So if one of the categories of the document is exactly the target category, the distance and also the category score for that target category are 0, no matter what other categories are assigned to the document.

Besides the category score, we also need a content score for each document. This score is calculated using a language model with Jelinek-Mercer smoothing without length prior. Finally, we combine the content and category scores through a linear combination. Both scores are calculated in the log space, and then a weighted addition is made.

$$S(d|QT) = \mu \cdot log(P(q|d)) + (1 - \mu) \cdot log(S_{cat}(d|QT)) \quad (1)$$

3. EXPERIMENTS

In this section we first describe our experimental setup, then examine the effects of using category information for retrieval, and compare the effects of manual and automatic category assignment.

3.1 Experimental Setup

To create our baseline runs incorporating only the content score, we use Indri [4]. Our baseline is a language model using Jelinek-Mercer smoothing with $\lambda = 0.1$. We also apply pseudo-relevance feedback, using the top 50 terms from the top 10 documents. The category score is only calculated for the top 1000 documents of the baseline run. These documents are reranked to produce the run that combines content and category score.

Our topic set consists of the INEX Ad hoc topics from 2007, to which we manually and automatically assigned categories. For the automatically assigned categories, we have two parameters, N the number of top results to use, and T the number of target categories that is assigned for each topic. For the parameter μ we tried values from 0 to 1, with steps of 0.1. The best values of μ turned out to be on the high end of this spectrum, therefore we added two additional values of μ: 0.95 and 0.98.

3.2 Experimental Results

To find the best values for parameters N and T, we compare the top categories in our run to the top categories of the relevant documents. Table 1 shows the overlap between the automatically assigned categories and the relevant categories.

The results of our experiments expressed in MAP are summarized in Table 2. This table gives the content score, which we use as our baseline, the category score, the combined score using $\mu = 0.9$ and the best score of their combination with the corresponding value of μ. The best value for μ differs per topic set, but for all sets μ is quite close to 1. The reason for the high μ values is that the category scores are an order of magnitude larger, because instead of scoring a few query terms, all the terms occurring in the language model of the category are scored. So even with small weights, the category score contributes significantly to the total score.

Table 2: Retrieval results in MAP

Cats		Category	Comb.	Best Score	
N	T	$\mu = 0.0$	$\mu = 0.9$	μ	
Baseline					0.3151
Manual		$0.1821^{\bullet}$	$0.3508^{\bullet}$	0.9	$0.3508^{\bullet}$
Top 10	1	$0.1695^{\bullet}$	0.3247^{-}	0.95	$0.3346^{\bullet}$
Top 20	1	$0.1759^{\bullet}$	0.3279^{-}	0.95	$0.3384^{\bullet}$
Top 10	2	$0.1879^{\bullet}$	0.3276^{-}	0.95	$0.3438^{\bullet}$
Top 20	2	$0.1912^{\bullet}$	0.3236^{-}	0.95	0.3411°

Significance of increase/decrease over baseline according to t-test, one-tailed, at significance levels 0.05($^{\circ}$), 0.01($^{\circ}$), and 0.001($^{\bullet}$).

When we use the category information for the ad hoc topics with manually assigned categories MAP improves significantly with an increase of 11.3%. Using the automatically assigned topics, almost the same results are achieved. The best automatic run uses the top 10 documents and takes the top 2 categories, reaching a MAP of 0.3438, a significant improvement of 9.1%.

4. CONCLUSION

In this paper we have investigated the use of categories to find information in Wikipedia. Using category information leads to significant improvements over the baseline, so we find a positive answer to our first research question. Considering our second research question, automatically assigned categories prove to be good substitutions for manually assigned target categories. Similar to the runs using manually assigned categories, using the automatically assigned categories leads to significant improvements over the baseline.

Acknowledgments

This research was supported by the Netherlands Organization for Scientific Research (NWO, under project # 612.-066.513, 639.072.601, and 640.001.501).

REFERENCES

[1] D. Hiemstra. *Using Language Models for Information Retrieval*. PhD thesis, Center for Telematics and Information Technology, University of Twente, 2001.

[2] D. Hiemstra, S. Robertson, and H. Zaragoza. Parsimonious language models for information retrieval. In *Proceedings SIGIR 2004*, pages 178–185. ACM Press, New York NY, 2004.

[3] R. Kaptein and J. Kamps. Web directories as topical context. In *Proceedings of the 9th Dutch-Belgian Workshop on Information Retrieval (DIR 2009)*, 2009.

[4] T. Strohman, D. Metzler, H. Turtle, and W. B. Croft. Indri: a language-model based search engine for complex queries. In *Proceedings of the International Conference on Intelligent Analysis*, 2005.

[5] A.-M. Vercoustre, J. Pehcevski, and J. A. Thom. Using wikipedia categories and links in entity ranking. In *Focused Access to XML Documents*, pages 321–335, 2007.