Supervised Query Modeling Using Wikipedia

Edgar Meij and Maarten de Rijke
ISLA, University of Amsterdam, The Netherlands
{edgar.meij, derijke}@uva.nl

ABSTRACT

We use Wikipedia articles to semantically inform the generation of query models. To this end, we apply supervised machine learning to automatically link queries to Wikipedia articles and sample terms from the linked articles to re-estimate the query model. On a recent large web corpus, we observe substantial gains in terms of both traditional metrics and diversity measures.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval

General Terms

Algorithms, Experimentation, Measurement

Keywords

Machine Learning, Query Modeling, Wikipedia

1. INTRODUCTION

In a web retrieval setting, there is a clear need for precision enhancing methods [5]. For example, the query "the secret garden" (a novel that has been adapted into movies and musicals) is a query that is easily led astray because of the generality of the individual query terms. While some methods address this issue at the document level, e.g., by using anchor texts or some function of the web graph, we are interested in improving the query; a prime example of such an approach is leveraging phrasal or proximity information [8]. Besides degrading the user experience, another significant downside of a lack of precision is its negative impact on the effectiveness of pseudo relevance feedback methods. An example of this phenomenon can be observed for a query such as "indexed annuity" where the richness of the financial domain plus the broad commercial use of the web introduces unrelated terms. To address these issues, we propose a semantically informed manner of representing queries that uses supervised machine learning on Wikipedia. We train an SVM that automatically links queries to Wikipedia articles which are subsequently used to update the query model.

Wikipedia and supervised machine learning have previously been used to select optimal terms to include in the query model [10]. We, however, are interested in selecting those Wikipedia articles which best describe the query and use those to sample terms from. This is similar to the unsupervised manner used, e.g., in the context of retrieving blogs [9]. Such approaches are completely unsupervised in that they only consider a fixed number of pseudo relevant Wikipedia articles. As we will see below, focusing this set using machine learning improves overall retrieval performance.

Copyright is held by the author/owner(s).
SIGIR'10, July 19–23, 2010, Geneva, Switzerland.
ACM 978-1-60558-896-4/10/07.

2. QUERY MODELING

We adopt a language modeling for IR framework in which documents are ranked according to their likelihood of generating the query: $\log P(D|Q) \propto \log P(D) + \sum_{t \in Q} P(t|\theta_Q) \log P(t|\theta_D)$. In our experiments we assume a uniform document prior and apply Bayesian smoothing using a Dirichlet prior (set to the average document length) to obtain each document model θ_D. For the query model we use a linear interpolation: $P(t|\theta_Q) = \lambda_Q P(t|\tilde{\theta}_Q) + (1 - \lambda_Q) P(t|\hat{\theta}_Q)$, where $P(t|\tilde{\theta}_Q)$ indicates the empirical estimate on the initial query and $P(t|\hat{\theta}_Q)$ an expanded part which we obtain using the formula below. Note that when we set $\lambda_Q = 1$ we obtain a query-likelihood ranking which will serve as our baseline.

We take relevance model 1 for our estimations of $P(t|\hat{\theta}_Q)$ [6]:

$$P(t|\hat{\theta}_Q) = \frac{1}{|R|} \sum_{D \in R} P(t|D) P(Q|D). \quad (1)$$

Here, R indicates a set of (pseudo) relevant documents which we obtain in three ways: (i) on the collection ("normal" pseudo relevance feedback), (ii) on Wikipedia (similar to so-called "external expansion" [4, 9]), and (iii) using automatically linked Wikipedia articles, which are introduced in the next section.

3. LINKING QUERIES TO WIKIPEDIA

To be able to derive query models based on Wikipedia, we first need to link queries to Wikipedia articles. To this end, we follow the approach in [7] which maps queries to DBpedia concepts, without performing any subsequent query modeling as we do in this paper. We take their best performing settings, i.e., SVM with a polynomial kernel using full queries. Instead of using a proprietary dataset, however, we take two ad hoc TREC test collections, i.e., TREC Terabyte 2004–2006 (.GOV2) and TREC Web 2009 (ClueWeb09, Category A).[1] In order to classify Wikipedia articles as being relevant to a query, the approach uses manual query-to-article annotations to train an SVM model. For new queries, a retrieval run is performed on Wikipedia which is then classified using the trained model. The output of this step is a binary classification on each Wikipedia article, where the class indicates the relevance status as predicted by the SVM.

Our features include those pertaining to the query, the Wikipedia article, and their combination. See [7] for an extensive description of each feature. Since we are using ad hoc test collections, we do not have session information and omit the history-based features used there. In order to obtain training data, we have asked 4 annotators to manually identify all relevant Wikipedia articles for each query. The average number of Wikipedia articles the annotators identified per query is around 2 for both collections. The average number of articles identified as relevant per query by SVM is slightly different, with 1.6 for TREC Terabyte and 2.7 for TREC

[1] http://trec.nist.gov/.

	λ_Q	MAP	MRR	Recall	P10
QL	1	0.2803	0.7121	7874	0.5081
RM (C)	0.5	**0.2882**	0.6126	7599	0.5068
RM (WP)	0.5	0.2680	**0.7331**	7364	0.5203
WP-SVM	0.8	0.2856	0.7108	**7902**	**0.5324**
WP-SVM	0.5	0.2769	0.6937	7731	0.5176
WP-SVM	0	0.2284	0.6307	6965	0.4392

Table 1: Results on the TREC Terabyte 2004–2006 collection.

	λ_Q	MAP	MRP	MPC30	MNDCG
QL	1	0.02583	0.07765	0.08333	0.04443
RM (C)	0.5	0.02523	0.07612	0.07823	0.04107
RM (WP)	0.5	0.02320	0.07274	0.07847	0.04359
WP-SVM	0.8	0.03371	0.08882	0.11304	0.06188
WP-SVM	0.5	**0.03635**	**0.08961**	**0.13437**	**0.07529**
WP-SVM	0	0.02917	0.07403	0.12577	0.06480

Table 2: Results on the TREC Web 2009 collection (using stat measures [2]).

	λ_Q	eMAP	α-NDCG@10	IA-P@10
QL	1	0.03614	0.04200	0.01700
RM (C)	0.5	0.03919	0.03200	0.01300
RM (WP)	0.5	0.03474	0.03900	0.01600
WP-SVM	0.8	0.04702	0.05700	0.03000
WP-SVM	0.5	0.06364	**0.06100**	**0.03500**
WP-SVM	0	**0.09418**	0.03300	0.01800

Table 3: Results on the TREC Web 2009 test collection (using expectedMAP (eMAP) and diversity measures [1–3]).

Web 2009. This seems to be due to the differences in queries; the TREC Web queries are shorter and, thus, more prone to ambiguity.

For the TREC Web 2009 query (#48) "wilson antenna," it predicts ROBERT WOODROW WILSON as the only relevant article, classifying articles such as MOUNT WILSON (CALIFORNIA) as not relevant. For the query "the music man" (#42) it identifies the company, song, 1962 film, and 2003 film which indicates the inherent ambiguity of many web queries. The same effect can be observed for the query "disneyland hotel" (#39) with articles TOKYO DISNEYLAND HOTEL, DISNEYLAND HOTEL (CALIFORNIA), and DISNEYLAND HOTEL (PARIS). There are also mistakes however, such as predicting the article FLAME OF RECCA (a Japanese manga series) for the query (#49) "flame designs."

4. RESULTS AND DISCUSSION

To determine whether the automatically identified articles are a useful resource to improve the query model, we compare our approach (WP-SVM) against a query-likelihood (QL) baseline and against Eq. 1 on pseudo relevant documents. In the latter case, we use either the collection (RM (C)) or the top-ranked Wikipedia articles (RM (WP)). For both we use the top 10 retrieved documents. In order to make results comparable, we include the 10 terms with the highest probability in $P(t|\hat{\theta}_Q)$ for all approaches. We leave the influence of varying these numbers for future work.

To train the SVM model, we split each test collection in a training and test set. For TREC Terabyte 2004–2006, we have 150 topics which are split equally. For TREC Web 2009 we have 50 topics and use five-fold cross validation.

Tables 1 and 2 show the results on TREC Terabyte and Web 2009 respectively (best scores in boldface). For TREC Terabyte, we observe that WP-SVM obtains highest recall and P10. Although pseudo relevance feedback on the collection obtains highest MAP, MRR is relatively low. An example of a topic helped by WP-SVM is "train station security measures" (#711) caused by the suggested article SECURITY ON THE MASS RAPID TRANSIT.

As to TREC Web 2009, performing pseudo relevance feedback on the collection introduces very general terms and thus does not improve overall retrieval effectiveness. Using WP-SVM to estimate the query model, however, introduces focused terms which improves overall performance. These results indicate that supervised query modeling using Wikipedia is helpful for large, noisy collections.

When we evaluate WP-SVM on the TREC Web 2009 collection using the diversity track's measures, cf. Table 3, we arrive at the same picture. Using WP-SVM we obtain an α-nDCG@10 score of 0.06100 which would have placed this run in the top-7 of participating systems in that particular track. This finding, in conjunction with the examples provided earlier, indicates that our query modeling approach caters for multiple interpretations of the query since prominent terms from each identified Wikipedia article are included in the query model.

5. CONCLUSIONS

We have presented a query modeling method based on Wikipedia that is aimed at obtaining high-precision representations of the original query. We find limited improvements on a relatively small web collection, only beating state-of-the-art query expansion methods according to some metrics. On a much larger web corpus, we achieve improvements on all metrics, whether precision or recall oriented. When using diversity measures, we observe major improvements, especially when relying exclusively on externally derived contributions to the query model.

Acknowledgements This research was supported by the European Union's ICT Policy Support Programme as part of the Competitiveness and Innovation Framework Programme, CIP ICT-PSP under grant agreement nr 250430, by the DuOMAn project carried out within the STEVIN programme which is funded by the Dutch and Flemish Governments under project nr STE-09-12, and by the Netherlands Organisation for Scientific Research (NWO) under project nrs 612.066.512, 612.061.814, 612.061.815, 640.004.802.

6. REFERENCES

[1] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong. Diversifying search results. In *WSDM '09*, 2009.
[2] B. Carterette, V. Pavlu, E. Kanoulas, J. A. Aslam, and J. Allan. Evaluation over thousands of queries. In *SIGIR '08*, 2008.
[3] C. L. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon. Novelty and diversity in information retrieval evaluation. In *SIGIR '08*, 2008.
[4] F. Diaz and D. Metzler. Improving the estimation of relevance models using large external corpora. In *SIGIR '06*, 2006.
[5] T. Joachims, L. Granka, B. Pan, H. Hembrooke, and G. Gay. Accurately interpreting clickthrough data as implicit feedback. In *SIGIR '05*, 2005.
[6] V. Lavrenko and B. W. Croft. Relevance models in information retrieval. In B. W. Croft and J. Lafferty, editors, *Language Modeling for Information Retrieval*. Kluwer, 2003.
[7] E. J. Meij, M. Bron, B. Huurnink, L. Hollink, and M. de Rijke. Learning semantic query suggestions. In *ISWC '09*, 2009.
[8] G. Mishne and M. de Rijke. Boosting web retrieval through query operations. In *ECIR '05*, 2005.
[9] W. Weerkamp, K. Balog, and M. de Rijke. A generative blog post retrieval model that uses query expansion based on external collections. In *ACL-ICNLP 2009*, 2009.
[10] Y. Xu, G. J. Jones, and B. Wang. Query dependent pseudo-relevance feedback based on wikipedia. In *SIGIR '09*, 2009.