The Hedge Algorithm for Metasearch at TREC 2007

Javed A. Aslam* Virgil Pavlu Olena Zubaryeva
College of Computer and Information Science
Northeastern University

Abstract

Aslam, Pavlu, and Savell [3] introduced the Hedge algorithm for metasearch which effectively combines the ranked lists of documents returned by multiple retrieval systems in response to a given query. It has been demonstrated that the Hedge algorithm is an effective technique for metasearch, often significantly exceeding the performance of standard metasearch and IR techniques over small TREC collections. In this work, we explore the effectiveness of Hedge as an *automatic* metasearch engine over the much larger GOV2 collection on about 1700 topics as part of the Million Query Track of TREC 2007.

1 Introduction

Aslam, Pavlu, and Savell introduced a unified framework for simultaneously solving the problems of metasearch, pooling, and system evaluation based on the Hedge algorithm for on-line learning [3]. Given the ranked lists of documents returned by a collection of IR systems in response to a given query, Hedge is capable of matching and often exceeding the performance of the best underlying retrieval system; given relevance feedback, Hedge is capable of "learning" how to optimally combine the input systems, yielding a level of performance which often significantly exceeds that of the best underlying system.

In previous experiments with smaller TREC collections [3], it has been shown that after only a handful of judged feedback documents, Hedge is able to significantly outperform the CombMNZ and Condorcet metasearch techniques. It has also been shown that Hedge is able to efficiently construct pools containing significant numbers of relevant documents and that these pools are highly effective at evaluating the underlying systems [3]. Although the Hedge algorithm has been shown to be a strong technique for metasearch, pooling, and system evaluation using the relatively small or moderate TREC collections (TRECs 3, 5, 6, 7, 8), it has yet to be demonstrated that the technique is scalable to corpora whose data size is at the terabyte level. In this work, we assess the performance of Hedge on a terabyte scale, presenting testing results using the Million Query Track topics and data.

Finally, we note that in the context of this TREC, in the absence of feedback, Hedge is a fully automatic metasearch algorithm.

2 Results

In the Million Query Track, Hedge was run with no feedback (ergo the name of the submission "hedge0") as an automatic metasearch engine. We indexed the collection using the Lemur Toolkit; that process took about 3 days using a 2-processor dual-core Opteron machine (2.4 GHz/core).

2.1 Underlying IR systems

The underlying systems include: (1) two tf-idf retrieval systems; (2) three KL-divergence retrieval models, one with Dirichlet prior smoothing, one with Jelinek-Mercer smoothing, and the last with absolute discounting; (3) a cosine similarity model; (4) the OKAPI retrieval model; (5) and the INQUERY retrieval method. All of the above retrieval models are provided as standard IR systems by the Lemur Toolkit [1]. For each query and retrieval system, we considered the top 1,000 scored documents for that retrieval system. Once all retrieval systems were run against all queries, we ran the Hedge algorithm [3] to perform metasearch on the ranked lists obtained. These models were run on 10,000 topics using the

*We gratefully acknowledge the support provided by NSF grants CCF-0418390 and IIS-0534482.

System	*MAP*	*p@20*
Jelinek-Mercer	0.2257	0.3780
Dirichlet	0.2100	0.4200
TFIDF	0.1993	0.4250
Okapi	0.1906	0.4270
log-TFIDF	0.1661	0.4140
Absolute Discounting	0.1575	0.3660
Cosine Similarity	0.0875	0.1960
CombMNZ	0.2399	0.4550
Condorcet	0.2119	0.4200
hedge0	0.2297	0.4260

Table 1: Results for input and metasearch systems on the Terabyte05 collection and topics. CombMNZ, Cordorcet, and hedge0 were run over all input systems.

statAP (1153 topics)	0.2175
statR-prec (1153 topics)	0.2266
statPrec@30 (1153 topics)	0.1728
EMAP (1700 topics)	0.0641

Table 2: Performance over MQ topics, where the evaluation is performed using the MQ evaluation methodology. StatAP, statR-prec, and statPrec@30 refer to estimates of average precision, R-precision, and precision@30, averaged over 1153 topics. EMAP refers to the MTC evaluation over 1700 topics; note that the EMAP value is not on the same scale as traditional MAP values [2].

GOV2 collection.

For reference, Table 1 illustrates that both hedge0 and CombMNZ are able to exceed the performance of the best underlying system (Terabyte05 data). This demonstrates that Hedge alone, even without any relevance feedback, is a successful metasearch technique, exceeding the metasearch performance of Condorcet and rivaling the performance of CombMNZ.

2.2 Results for Million Query 07

Tables 3 & 4 and Figure 1 present the performance of hedge0 on the 2007 Million Query Track collection and topics, separately for the MQ evaluation methods and for the 149 Terabyte topics using traditional evaluation methods and metrics. This performance was in line with expectations and previous results.

MAP	0.1708
R-prec	0.2411
bpref	0.2414
recip-rank	0.6039
retrieved	135075
relevant	26917
relevant retrieved	13944

Table 3: Performance over the 149 Terabyte06 topics, where the evaluation was performed using traditional methods and metrics.

Precision at Recall (149 Terabyte06 topics)	
recall	precision
.00	0.6611
.10	0.3748
.20	0.3096
.30	0.2499
.40	0.2015
.50	0.1558
.60	0.1019
.70	0.0592
.80	0.0351
.90	0.0086
1.00	0.0035

Precision at Rank (149 Terabyte06 topics)	
rank	precision
at 5 docs	0.4174
at 10 docs	0.3826
at 15 docs	0.3575
at 20 docs	0.3453
at 30 docs	0.3327
at 100 docs	0.2644
at 200 docs	0.2209
at 500 docs	0.1473
at 1000 docs	0.0936
R-precision	0.2411

Table 4: Performance over the 149 Terabyte06 topics, where the evaluation was performed using traditional methods and metrics.

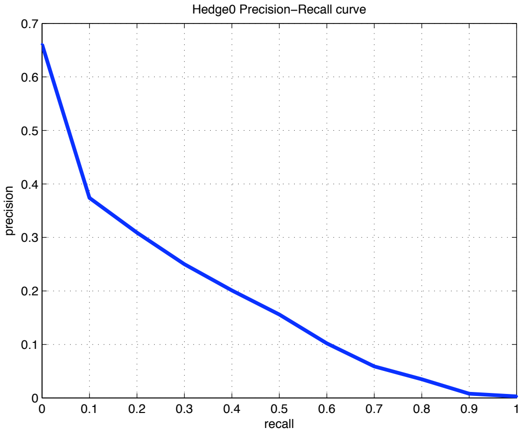

Figure 1: Hedge0: Precision-recall curve averaged over the 149 Terabyte06 topics.

References

[1] The lemur toolkit for language modeling and information retrieval. http://www.cs.cmu.edu/~lemur.

[2] James Allan, Ben carterette, Javed A. Aslam, Virgil Pavlu, Blagovest Dachev, and Evangelos Kanoulas. The Million Query Track 2007 Overview. In *Proceedings of the Sixteenth Text REtrieval Conference (TREC 2007)*, 2008.

[3] Javed A. Aslam, Virgiliu Pavlu, and Robert Savell. A unified model for metasearch, pooling, and system evaluation. In Ophir Frieder, Joachim Hammer, Sajda Quershi, and Len Seligman, editors, *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, pages 484–491. ACM Press, November 2003.