SPUD out of the Box: An Implementation and Evaluation in Lucene 5.0.0

Ronan Cummins

Abstract

We present details regarding the implementation and evaluation of the SPUD language model in Lucene 5.0.0. The implementation of the SPUD approach and all the code needed to re-run these experiments is available at `http://ir.dcs.gla.ac.uk/~ronanc/spud/spud.tar`.

1 Retrieval Approaches and Hyper-Parameters

Lucene already contains implementations for BM25 [3] and the multinomial language model using Dirichlet priors (Dir) [4][1]. As the SPUD retrieval method [2] makes use of different length normalisation characteristics, re-indexing of document collections is required to gather these measures. Table 1 shows the default parameters used for each retrieval method. Note that no effort was spent tuning these parameters as we wish to focus on an *out-of-the-box* comparison of the retrieval methods (i.e. we feel this is one of the main reason for the development of unsupervised approaches to retrieval).

Table 1: Default Hyper-Parameters

BM25	$k_1 = 1.2$, $b = 0.75$
Dir	$\mu = 2000$
SPUD	$\omega = 0.8$

It is worth noting that the SPUD approach also calculates an approximation of the mass of the background DCM (Pólya) distribution before retrieval. This is implemented in our code but the value is currently not stored in the index (and ideally it should be). This value is collection-specific and only needs to be calculated once per collection before running any queries (by default it does this).

[1] From inspection, Lucene's implementation of Dir does not seem to apply document length normalisation in the same way as outlined in [4]. Therefore, we also report results for a re-implementation of the approach. Lucene's implementation of BM25 seems to be correct.

2 Results

Table 1 shows the default retrieval effectiveness in terms of mean average precision and NDCG@10 (as per [1]) of SPUD (our implementation), BM25 (as implemented in Lucene 5.0.0), Dir (as implemented in Lucene 5.0.0), and Dir′ (our re-implementation) on a number of TREC[2] test collections. We tested title only, description only, and narrative only queries.

Table 2: Mean Average Precision (NDCG@10) of Retrieval Methods Implemented in Lucene with Default parameters

Collection	Robust-04	WT2G	Gov2
# docs	0.5M	0.M	25M
Topics	301-450 601-700	401-450	701-850
title only			
BM25	0.232 (0.446)	0.242 (0.420)	0.255 (0.530)
Dir	0.241 (0.446)	0.300 (0.490)	0.284 (0.558)
Dir′	0.247 (0.455)	0.314 (0.494)	0.293 (0.563)
SPUD	**0.259 (0.478)**	**0.315 (0.497)**	**0.316 (0.596)**
description only			
BM25	0.228 (0.450)	0.236 (0.437)	0.220 (0.512)
Dir	0.230 (0.429)	0.264 (0.423)	0.224 (0.491)
Dir′	0.242 (0.447)	0.260 (0.446)	0.218 (0.481)
SPUD	**0.253 (0.470)**	**0.283 (0.475)**	**0.241 (0.515)**
narrative only			
BM25	0.266 (0.509)	0.266 (0.486)	0.295 (0.630)
Dir	0.251 (0.471)	0.278 (0.472)	0.274 (0.580)
Dir′	0.262 (0.491)	0.271 (0.466)	0.294 (0.599)
SPUD	**0.295 (0.533)**	**0.295 (0.503)**	**0.317 (0.632)**

3 Summary

The comparison of retrieval methods based on a open-source framework (Lucene 5.0.0) has shown that the SPUD approach is a highly effective approach for many types of queries. We note that the actual results reported here are different from those in [2] as the original experiments were not run under Lucene.

References

[1] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In

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

Proceedings of the 22nd international conference on Machine learning, pages 89–96. ACM, 2005.

[2] Ronan Cummins, Jiaul H. Paik, and Yuanhua Lv. A pólya urn document language model for improved information retrieval. *CoRR*, abs/1502.00804, 2015.

[3] S. E. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, pages 232–241, New York, NY, USA, 1994. Springer-Verlag New York, Inc.

[4] Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions of Information Systems*, 22:179–214, April 2004.