Cumulative Progress in Language Models for Information Retrieval

Antti Puurula
The University of Waikato
Private Bag 3105
Hamilton 3240, New Zealand
asp12@students.waikato.ac.nz

Abstract

The improvements to ad-hoc IR systems over the last decades have been recently criticized as illusionary and based on incorrect baseline comparisons. In this paper several improvements to the LM approach to IR are combined and evaluated: Pitman-Yor Process smoothing, TF-IDF feature weighting and model-based feedback. The increases in ranking quality are significant and cumulative over the standard baselines of Dirichlet Prior and 2-stage Smoothing, when evaluated across 13 standard ad-hoc retrieval datasets. The combination of the improvements is shown to improve the Mean Average Precision over the datasets by 17.1% relative. Furthermore, the considered improvements can be easily implemented with little additional computation to existing LM retrieval systems. On the basis of the results it is suggested that LM research for IR should move towards using stronger baseline models.

1 Introduction

Research on ad-hoc Information Retrieval (IR) has been recently criticized for being based on incorrect baseline comparisons. According to extensive evaluation of IR systems from over a decade, no progress has been demonstrated on standard datasets (Armstrong et al., 2009a; Armstrong et al., 2009b).

In this paper we propose that although much of this criticism is valid, much of the more recent progress in Language Model-based (LM) IR has not been evaluated or received the attention that it deserved. We evaluate on 13 standard IR datasets some of the improvements that have been suggested to LMs over the years. It is shown that the combination of Pitman-Yor Process smoothing, TF-IDF feature weighting and Model-based Feedback produces a substantial and cumulative improvement over the common baseline LM smoothing methods.

2 Improvements to LMs for IR

2.1 LM Approach to IR

The LM approach to ad-hoc IR considers documents and queries to be generated by underlying n-gram LMs. The Query Likelihood (QL) framework for LM retrieval (Hiemstra, 1998) treats queries as being generated by document models, reducing the retrieval of the most relevant documents into ranking documents by the posterior probability of each document given the query. Unigram LMs and a uniform distribution over document priors is commonly assumed, so that the QL-score for each document correspond to the conditional log-probability of the query given the document:

$$\log p_m(\boldsymbol{w}) = \log Z(\boldsymbol{w}) + \sum_n w_n \log p_m(n), \quad (1)$$

where $Z(\boldsymbol{w})$ is a Multinomial normalizer, $\boldsymbol{w}$ is the query word count vector, and $p_m(n)$ is given by a Multinomial estimated from the document word count vector $\boldsymbol{d}_m$:

$$p_m(n) = \frac{d_{mn}}{||\boldsymbol{d}_m||_1} \quad (2)$$

The QL framework is the standard application of LMs to IR. It is equivalent to using a Multinomial Naive Bayes model for ranking, with classes corresponding to documents, and a uniform prior over the document models.

Antti Puurula. 2013. Cumulative Progress in Language Models for Information Retrieval. In *Proceedings of Australasian Language Technology Association Workshop*, pages 96−100.

2.2 Pitman-Yor Process Smoothing

The standard choices for LM model smoothing in IR have been Dirichlet Prior (DP) and 2-stage Smoothing (2SS) (Zhai and Lafferty, 2004; Smucker and Allan, 2007; Zhai, 2008). A recent improvement has been Pitman-Yor Process (PYP) smoothing, derived as approximate inference on a Hierarchical Pitman-Yor Process (Momtazi and Klakow, 2010; Huang and Renals, 2010). All methods interpolate document model parameter estimates linearly with a background model, differing in how the interpolation weight is determined. PYP applies additionally power-law discounting of the document counts. For all methods the smoothed parameter estimates can be expressed in the form:

$$p_m(n) = (1 - \alpha_m)\frac{d'_{mn}}{||\boldsymbol{d}'_m||_1} + \alpha_m p^c(n), \quad (3)$$

where $\boldsymbol{d}'_m$ is the discounted count vector, $p^c(n)$ is the background model and α_m is the smoothing weight.

DP chooses the smoothing weight as $\alpha_m = 1 - \frac{||\boldsymbol{d}_m||_1}{||\boldsymbol{d}_m||_1+\mu}$, where μ is a parameter. 2SS combines DP with Jelinek-Mercer smoothing, using $\alpha_m = 1 - \frac{||\boldsymbol{d}_m||_1-\beta||\boldsymbol{d}_m||_1}{||\boldsymbol{d}_m||_1+\mu}$, where β is a linear interpolation parameter. PYP uses $\alpha_m = 1 - \frac{||\boldsymbol{d}'_m||_1}{||\boldsymbol{d}_m||_1+\mu}$, with the discounted counts $d'_{mn} = \max(d_{mn} - \Delta_{mn}, 0)$, where $\Delta_{mn} = \delta d^{\delta}_{mn}$ is produced by Power-law Discounting (Huang and Renals, 2010) with the discounting parameter δ. Replacing the discounting in PYP with the linear Jelinek-Mercer smoothing reproduces the 2SS estimates: $||\boldsymbol{d}'_m||_1 = ||\boldsymbol{d}_m||_1 - \beta||\boldsymbol{d}_m||_1$. PYP is therefore a non-linear discounting version of 2SS.

The background model $p^c(n)$ is commonly a collection model estimated by treating all available documents as a single large document: $p^c(n) = \sum_m \frac{d_{mn}}{\sum_{n'}\sum_{m'} d_{m'n'}}$. A uniform distribution is less commonly used: $p^c(n) = \frac{1}{|N|}$.

2.3 TF-IDF Feature Weighting

Unigram LMs make several incorrect modeling assumptions about natural language, such as considering all words equally informative. Feature weighting has shown to be useful in improving the effectiveness of Multinomial models in both IR (Smucker and Allan, 2006; Momtazi et al., 2010) and other uses (Rennie et al., 2003; Frank and Bouckaert, 2006). This is in contrast to earlier theory in IR that considered smoothing with collection model as non-complementary to feature weighting (Zhai and Lafferty, 2004).

TF-IDF word weighting for dataset documents can be done by:

$$d_n = \log(1 + \frac{d''_n}{||\boldsymbol{d}''||_0}) \log \frac{M}{M_n}, \quad (4)$$

where $\boldsymbol{d}''$ is the unweighted count vector, $||\boldsymbol{d}''||_0$ the number of unique words in the document, M the number of documents and M_n the number of documents where the word n occurs.

The first factor in Equation 4 is a TF log transform, using unique length normalization (Singhal et al., 1996). The second factor is Robertson-Walker IDF (Robertson and Zaragoza, 2009). Weighting query word vectors works identically. Collection model smoothing has an overlapping function to IDF weighting (Hiemstra and Kraaij, 1998). Here this interaction is taken into account by changing the background smoothing distribution into a uniform distribution.

2.4 Feedback Models

Pseudo-feedback is a traditional method used in IR that can have a large impact on retrieval performance. The top ranked documents can be used to construct a query model for a second pass of retrieval. With LMs there are two different ways to formalize this: KL-divergence Retrieval (Zhai and Lafferty, 2001) and Relevance Models (Lavrenko and Croft, 2001). Both methods enable replacing the query vector with a model (Zhai, 2008).

A number of variants exist for LM feedback modeling. Practical modeling choices are using only the top K retrieved documents, and truncating the query model to the words present in the original query (Zhai, 2008). The documents can be weighted according to the posterior probability of the document given the query, $p(\boldsymbol{d}_m|\boldsymbol{w}) \propto p_m(\boldsymbol{w})$ (Lavrenko and Croft, 2001).

The query model can also be interpolated linearly with the original query (Zhai and Lafferty, 2001). These modeling choices are combined here, resulting in a robust feedback model that has the same complexity for inference as the original query.

Using the top $K = 50$ retrieved documents, the query words $w_n > 0$ can be interpolated with the top document models $p_k(n)$:

$$w_n = (1 - \lambda)\frac{w'_n}{||\boldsymbol{w}'||_1} \lambda \sum_k \frac{p_k(\boldsymbol{w}')\, p_k(n)}{Z}, \quad (5)$$

where $\boldsymbol{w}'$ is the original query, λ is the interpolation weight, and Z is a normalizer for the feedback counts: $Z = \sum_{n:w'_n>0} \sum_k p_k(\boldsymbol{w}')p_k(n)$.

2.5 Experiments

Combining the LM improvements was evaluated on standard ad-hoc IR datasets. These are the TREC 1-5[1] datasets split according to data sources, OHSU-TREC[2] and FIRE 2008-2011[3]. Each dataset was filtered by stopwording, short word removal and Porter-stemming. The datasets were each split into a development set for calibrating parameters and a held-out evaluation set. The OHSU-TREC dataset was split according to documents, using ohsumed.87 for development and ohsumed.88-91 for evaluation. The TREC and FIRE datasets were split according to queries, using the first $3/5$ of queries for each year as development data and the remaining $2/5$ as the evaluation data. For OHSU-TREC the queries consisted of the title and description sections of queries 1-63. For TREC and FIRE the description sections were used from queries 1-450 and 26-175, respectively. Table 1 summarizes the dataset split sizes.

The software used for the experiments was SGMWeka version 1.44, an open source toolkit for generative modeling[4]. Ranking effectiveness for the experiments was evaluated using Mean Average Precision from the top 50 documents (MAP@50). Smoothing parameters were optimized for MAP@50 using a parallelized Gaussian random search algorithm (Luke, 2009) on the development sets. The significance of experiment results was tested on the evaluation set MAP@50 scores of each dataset, using paired one-sided t-tests, with significance level $p < 0.05$.

Table 1: Dataset documents, test queries

Data	Development		Evaluation	
	Docs	Test	Docs	Test
fire_en	21919	90	16075	60
ohsu_trec	36890	63	196555	63
trec_ap	47172	118	33474	80
trec_cr	5063	38	4006	29
trec_doe	10053	28	7717	10
trec_fbis	23207	68	17315	48
trec_fr	25185	112	20581	75
trec_ft	41452	113	30549	75
trec_la	25944	87	17834	56
trec_pt	1635	9	1792	5
trec_sjmn	9160	29	6469	19
trec_wsj	21847	60	15839	41
trec_zf	19901	60	13763	39

The experiment results are shown in Table 2. Comparing PYP to DP and 2SS, PYP improves significantly on DP smoothing. The difference to 2SS is considerable as well, but not statistically significant due to variance. Adding TF-IDF (+TI) weighting to PYP, the improvement becomes significant over the 2SS baseline. Adding feedback (+FB) results in an improvement that is significant compared to both other improvements. The overall mean improvement over 2SS is 4.07 MAP@50, a 17.1% relative improvement.

2.6 Discussion

This paper presented an empirical evaluation of combining improvements to information retrieval language models. Experiments on standard ad-hoc IR datasets show that several improvements significantly and cumulatively improve on the baseline methods of LM retrieval using 2SS and DP smoothing methods. This contrasts with the reported illusionary improvements in IR literature (Armstrong et al., 2009a; Armstrong et al., 2009b). The considered improvements require very little additional computation and can be implemented with small modifications to existing IR search engines.

[1] http://trec.nist.gov/data/test_coll.html
[2] http://trec.nist.gov/data/t9_filtering.html
[3] http://www.isical.ac.in/~clia/
[4] http://sourceforge.net/projects/sgmweka/

Table 2: Ranking effectiveness as % MAP@50.

Dataset	DP	2SS	PYP	PYP +TI	PYP +TI +FB
fire_en	44.44	44.46	45.16	44.68	48.04
ohsu_trec	29.73	29.72	28.77	31.24	32.33
trec_ap	22.76	23.05	24.41	24.91	28.55
trec_cr	17.03	17.17	18.02	17.88	19.47
trec_doe	26.49	24.97	30.58	30.98	34.66
trec_fbis	23.51	23.57	24.66	26.14	28.81
trec_fr	18.42	18.53	18.72	18.86	19.68
trec_ft	23.26	23.55	24.65	23.73	24.80
trec_la	18.05	19.27	19.06	20.43	20.78
trec_pt	13.23	11.57	11.64	22.45	27.53
trec_sjmn	20.84	21.47	20.27	16.83	17.12
trec_wsj	32.00	32.44	33.77	34.53	38.41
trec_zf	17.92	18.48	17.54	19.52	20.97
mean	23.67	23.71	24.40	25.55	27.78

Several LM improvements have also been developed that require considerable additional computation. Methods such as document neighborhood smoothing, passage-based language models, word correlation models and bigram language models have all been shown to substantially improve LM performance (Miller et al., 1999; Song and Croft, 1999; Clinchant et al., 2006; Krikon and Kurland, 2011). Unfortunately, like the improvements discussed in this paper, many of these methods lack publicly available implementations, have been pursued by few researchers, and have been evaluated on a limited number of datasets. Evaluation of methods such as these could yield practical tools for IR and other applications of LMs.

The criticism of progress in ad-hoc IR (Armstrong et al., 2009a; Armstrong et al., 2009a; Trotman and Keeler, 2011) has missed valuable developments in LM-based IR. A second matter neglected in this criticism is the shift towards the learning-to-rank framework of IR (Joachims, 2002; Li, 2011), where individual retrieval models have reduced roles as base rankers and features. In this context it is not necessary for models to improve on a single measure or replace older ones; rather, it is sufficient that new models provide complementary information for combination of results.

The work reported here is preliminary and further experiments are required to understand possible interaction effects between the combined improvements. Given the performance and simplicity of the evaluated improvements, the commonly used DP and 2SS baselines for LMs should not generally be used as primary baselines for IR experiments. The combination of improvements shown in this paper is one potential baseline.

References

Timothy G. Armstrong, Alistair Moffat, William Webber, and Justin Zobel. 2009a. Has adhoc retrieval improved since 1994? In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 692–693, New York, NY, USA. ACM.

Timothy G. Armstrong, Alistair Moffat, William Webber, and Justin Zobel. 2009b. Improvements that don't add up: ad-hoc retrieval results since 1998. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 601–610, New York, NY, USA. ACM.

Stéphane Clinchant, Cyril Goutte, and Eric Gaussier. 2006. Lexical entailment for information retrieval. In *Proceedings of the 28th European conference on Advances in Information Retrieval*, ECIR'06, pages 217–228, Berlin, Heidelberg. Springer-Verlag.

Eibe Frank and Remco R. Bouckaert. 2006. Naive bayes for text classification with unbalanced classes. In *Proceedings of the 10th European conference on Principle and Practice of Knowledge Discovery in Databases*, PKDD'06, pages 503–510, Berlin, Heidelberg. Springer-Verlag.

Djoerd Hiemstra and Wessel Kraaij. 1998. Twenty-one at trec-7: Ad-hoc and cross-language track. In *In Proc. of Seventh Text REtrieval Conference (TREC-7*, pages 227–238.

Djoerd Hiemstra. 1998. A linguistically motivated probabilistic model of information retrieval. In *Research and Advanced Technology for Digital Libraries*, volume 1513 of *Lecture Notes in Computer Science*, pages 569–584, Berlin, Germany. Springer Verlag.

Songfang Huang and Steve Renals. 2010. Power law discounting for n-gram language models. In *ICASSP*, pages 5178–5181. IEEE.

Thorsten Joachims. 2002. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 133–142, New York, NY, USA. ACM.

Eyal Krikon and Oren Kurland. 2011. A study of the integration of passage-, document-, and cluster-based information for re-ranking search results. *Inf. Retr.*, 14(6):593–616, December.

Victor Lavrenko and W. Bruce Croft. 2001. Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '01, pages 120–127, New York, NY, USA. ACM.

Hang Li. 2011. A short introduction to learning to rank. *IEICE Transactions*, 94-D(10):1854–1862.

Sean Luke. 2009. *Essentials of Metaheuristics*. Lulu, version 1.2 edition. Available for free at http://cs.gmu.edu/∼sean/book/metaheuristics/.

David R. H. Miller, Tim Leek, and Richard M. Schwartz. 1999. A hidden markov model information retrieval system. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '99, pages 214–221, New York, NY, USA. ACM.

Saeedeh Momtazi and Dietrich Klakow. 2010. Hierarchical Pitman-Yor language model for information retrieval. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 793–794, New York, NY, USA. ACM.

Saeedeh Momtazi, Matthew Lease, and Dietrich Klakow. 2010. Effective term weighting for sentence retrieval. In *Proceedings of the 14th European conference on Research and advanced technology for digital libraries*, ECDL'10, pages 482–485, Berlin, Heidelberg. Springer-Verlag.

Jason D. Rennie, Lawrence Shih, Jaime Teevan, and David R. Karger. 2003. Tackling the poor assumptions of naive bayes text classifiers. In *ICML'03*, pages 616–623.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3:333–389, April.

Amit Singhal, Chris Buckley, and Mandar Mitra. 1996. Pivoted document length normalization. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '96, pages 21–29, New York, NY, USA. ACM.

Mark D. Smucker and James Allan. 2006. Lightening the load of document smoothing for better language modeling retrieval. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pages 699–700, New York, NY, USA. ACM.

Mark D. Smucker and James Allan. 2007. An Investigation of Dirichlet Prior Smoothings Performance Advantage. Technical report, Department of Computer Science, University of Massachusetts, Amherst.

Fei Song and W. Bruce Croft. 1999. A general language model for information retrieval. In *Proceedings of the eighth international conference on Information and knowledge management*, CIKM '99, pages 316–321, New York, NY, USA. ACM.

Andrew Trotman and David Keeler. 2011. Ad hoc ir: not much room for improvement. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR '11, pages 1095–1096, New York, NY, USA. ACM.

Chengxiang Zhai and John Lafferty. 2001. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the tenth international conference on Information and knowledge management*, CIKM '01, pages 403–410, New York, NY, USA. ACM.

Chengxiang Zhai and John Lafferty. 2004. A study of smoothing methods for language models applied to information retrieval. *ACM Trans. Inf. Syst.*, 22(2):179–214, April.

ChengXiang Zhai. 2008. Statistical language models for information retrieval a critical review. *Found. Trends Inf. Retr.*, 2(3):137–213, March.