Fixed-Cost Pooling Strategies Based on IR Evaluation Measures

Aldo Lipani[1(✉)], Joao Palotti[1], Mihai Lupu[1], Florina Piroi[1], Guido Zuccon[2], and Allan Hanbury[1]

[1] Institute of Software Technology and Interactive Systems, TU Wien, Vienna, Austria
{aldo.lipani,joao.palotti,mihai.lupu,florina.piroi, allan.hanbury}@tuwien.ac.at

[2] Faculty of Science and Engineering, Queensland University of Technology, Brisbane, Australia
g.zuccon@qut.edu.au

Abstract. Recent studies have reconsidered the way we operationalise the pooling method, by considering the practical limitations often encountered by test collection builders. The biggest constraint is often the budget available for relevance assessments and the question is how best – in terms of the lowest pool bias – to select the documents to be assessed given a fixed budget. Here, we explore a series of 3 new pooling strategies introduced in this paper against 3 existing ones and a baseline. We show that there are significant differences depending on the evaluation measure ultimately used to assess the runs. We conclude that adaptive strategies are always best, but in their absence, for top-heavy evaluation measures we can continue to use the baseline, while for P@100 we should use any of the other non-adaptive strategies.

1 Introduction

Information Retrieval (IR) research relies heavily on well grounded empirical experiments that demonstrate the impact and merits of new techniques. The common framework of IR experimentation relies on the Cranfield paradigm [6,22] of a test collection (a collection of documents, a set of topics, and a set of relevance assessments); this paradigm has predominantly driven the study and comparison of IR systems' effectiveness in the last decades of IR research.

In the first Cranfield experiment, relevance was modelled as a complete relation, i.e. a relevance judgement was expressed for each topic-document pair in the collection. However the large increase in size of document collections and the costs involved in obtaining relevance judgements soon rendered it impossible to source judgements for every topic-document pair in the collection. Even for a relatively small test collection with half a million documents (i.e. far from web-scale) and a few tens of topics, the effort to create a complete set of relevance

This research was partly funded by the Austrian Science Fund (FWF) project number P25905-N23 (ADmIRE).

J.M. Jose et al. (Eds.): ECIR 2017, LNCS 10193, pp. 357–368, 2017.
DOI: 10.1007/978-3-319-56608-5_28

judgements would take more than a researcher's entire (hopefully long) lifetime. Until today, the most used method to avoid complete assessment is *pooling*.

The pooling method reduces the number of relevance judgements that are necessary in order to accurately assess the effectiveness of an IR system, or, more importantly, establishing the difference between the effectiveness of two systems. Pooling has been first introduced in the '70s [10], but has been used regularly only since the '90s with standardized IR benchmarking at the Text Retrieval Conference (TREC) [22]. Central to the use of pooling is that sufficiently many and sufficiently diverse systems have contributed to the creation of the *pool*, i.e. the set of documents that are collected for judgement. The most common and simplest pooling strategy is *Depth@k*, which prescribes the collection of relevance assessments only for the top k (referred to as the *pool depth*) documents from each of the document rankings of a number of IR systems.

Pooling, though used frequently to build test collections, was soon taken under scrutiny as it was observed that when the number of systems contributing to the pool was too low or the systems were not diverse enough, the identified set of relevant documents was not sufficient to reliably and accurately assess the effectiveness of an IR system that did not contribute to the pool [19]. This issue challenges the *re-usability of a test collection* as a tool for evaluating and comparing IR systems beyond those systems that contributed to the pool [21].

This test collection *bias* towards advantaging systems that participated in the pool creation over those that did not is ultimately due to an incomplete set of relevance judgements [11]. Zobel [26] first and Buckley et al. [2] later have shown that small test collections typically exhibit little bias, while large collections, such as modern web scale test collections, are affected by larger bias and thus such test collections may be rendered void when evaluating IR systems (especially those that did not contribute to the pool) if this bias is not controlled.

Research on controlling for pool bias follows two main approaches. On one hand there is work to reduce the bias at the test collection creation time. This has been done by devising alternative pooling strategies [4,14–16]. On the other hand, when the objective is the reuse of an existing test collection, research has explored the possibility of adjusting evaluation measures such that new systems can be fairly compared to the ones that contributed to the pool creation [12,23]. When the two approaches are combined, a new pooling strategy emerges, along with a matching evaluation measure [1,24], complying with the observation that performance measures are an intrinsic part of test collections [16].

This paper explores a family of strategies based on IR evaluation measures to identify documents to be placed in a pool of fixed size N, where the size is defined by a fixed budget, such that the test collection can be reliably used in later retrieval experiments. These strategies are: a baseline, $Take@N$; 3 pooling strategies as introduced by Moffat et al. [16], $RBPABased@N$, $RBPBBased@N$, and $RBPCBased@N$; and 3 newly proposed pooling strategies, $DCGBased@N$, $RRFBased@N$, and $PPBased@N$. These pooling strategies are empirically evaluated with respect to their impact on three common evaluation measures; the results are compared on a set of 11 TREC test collections.

2 Pooling Strategies

Our aim is to empirically study several strategies inspired by IR evaluation measures. In the following M denotes the function that associates a score to a given document d, retrieved by at least one run in the set of pooled runs R_p. The definition of M varies depending on the pooling strategy used and will be detailed in this section. The function $\rho(d, r)$ expresses the position (also called rank) of the document d in the run r.

The first fix-cost pooling strategy we present, $Take@N$, is also used as a baseline in the following experiments, similarly to previous study [15]. This strategy is based on the common $Depth@k$ pooling strategy, using the highest rank at which documents have been retrieved in the pooled runs to select the top N documents to assess. The strategies we present following $Take@N$ share the intuition behind it, replacing the choice by the mere document rank with the choice by a score, which is also function of the document rank. That is, the pooling strategies accumulate evidence of the importance of a document d for a given query based on both a) the rank $\rho(d, r)$ at which d has been retrieved in the pooled run $r \in R_p$, and b) on the particularities of a selection of evaluation measures. We describe now, in more detail, each of the pooling strategies with which we experiment in this paper.

Take@N (strategy T) creates, for each query, a global ranked list with the highest rank at which a retrieved document occurs in the R_p runs. The top N ranked documents for the query are selected into the pool. The $Take@N$ strategy is specified by the following definition for M:

$$\mathrm{M}(d, R_p) = \max_{r \in R_p} (-\rho(d, r)) \tag{1}$$

This pooling strategy blindly takes into consideration the contribution of all pooled runs, whether they provide relevant documents or not. This behaviour is also the most fair among the pooling strategies, guaranteeing that every pooled run will have almost the same number of documents selected for assessment (the difference in the number of selected documents between runs is maximum 1).

DCGBased@N (strategy DCG) uses the discount function defined in the discounted cumulative gain to rank candidate documents to pool [9]. The discount is characterized by an inverse log_2 decay function and a gain value of 1. Formally documents for pooling are ranked in decreasing order by the values computed by M, where:

$$\mathrm{M}(d, R_p) = \sum_{r \in R_p : d \in r} \mathrm{DCG}(\rho(d, r)) = \sum_{r \in R_p : d \in r} \frac{1}{\log_2(\rho(d, r))} \tag{2}$$

RRFBased@N (strategy RRF) is rooted in the reciprocal rank (RR) evaluation measure, which is commonly used to assess system effectiveness in tasks such as known item search, question answering, or query auto completion [8]. A variant of RR, the reciprocal rank fusion (RRF), has been used in data fusion [7]. RRF makes use of an additional parameter, α, that controls the decay

of the document contribution score as a function of rank. In this pooling strategy we employ the same idea, with $\alpha = 60$ as in Cormack et al. [7]; other values will be investigated in future work. Formally, candidate documents for the pool are ranked in decreasing order by the values computed with M where:

$$\mathrm{M}(d, R_p) = \sum_{r \in R_p : d \in r} \mathrm{RRF}(\rho(d, r)) = \sum_{r \in R_p : d \in r} \frac{1}{\rho(d, r) + \alpha} \tag{3}$$

PPBased@N (strategy PP, for *perfect precision*) is inspired by the family of measures that counts the number of relevant documents found at rank k divided by the number of documents up to rank k. Average Precision [3] and Sakai's Q-Measure [20] are examples of metrics belonging to this family. Since we model these measures as if all documents up to rank k were relevant, the rank score attributed to a document retrieved by runs in R_p is the number of runs that have retrieved that document:

$$\mathrm{M}(d, R_p) = \sum_{r \in R_p : d \in r} \mathrm{PP} = \sum_{r \in R_p : d \in r} 1 \tag{4}$$

This translates to a majority voting procedure to rank documents and select the top N.

RBPABased@N (strategy $RBPA$) computes pool document scores based on Rank Biased Precision (RBP) [17]. The RBP formula is characterized by a parameter p that models the user persistence, i.e. the likelihood that the user examines a document. The persistence parameter is effectively used to discount the contribution of a relevant document, similarly to other gain-discount based measures [5]. Pool candidate documents are ranked in decreasing order of the score computed by:

$$\mathrm{M}(d, R_p) = \sum_{r \in R_p : d \in r} \mathrm{RBPA}(\rho(d, r)) = \sum_{r \in R_p : d \in r} (1 - p) p^{\rho(d,r)-1} \tag{5}$$

In our experiments we use $p = 0.8$; this is akin to previous work that relied on RBP for evaluation [18,25] and for pooling [15,16]. The use of RBP as a document discount factor in weighting the contribution of documents to the pool creates a family of pooling strategies which, besides $RBPABased@N$, include $RBPBBased@N$ and $RBPCBased@N$ [16]. We next present the latter two.

RBPBBased@N (strategy $RBPB$) is an adaptive version of $RBPA$, which adds documents to the pool in an incremental way. By this strategy, for each run $r \in R_p$, we compute its residual $e(r)$, i.e. a value proportional to the number of not judged documents in the run. The residual is defined as:

$$e(r) = (1 - p) \sum_{d \in r : j(d) = ?} p^{\rho(d,r)-1} \tag{6}$$

where $j(d)$ is 1 if the document d is judged relevant, 0 if judged as not relevant, and ? If the document is not judged.

With each new judgement the score $M(d, R_p)$ is recomputed as the runs' residuals have clearly changed (thus the adaptive nature of $RBPBBased@N$); this means recomputing the score:

$$M(d, R_p) = \sum_{r \in R_p : d \in r} RBPB(\rho(d, r)) = \sum_{r \in R_p : d \in r} (1-p)p^{\rho(d,r)-1} \cdot e(r) \quad (7)$$

RBPCBased@N (strategy $RBPC$) is the second adaptive pooling strategy we present in this paper that uses both the RBP residuals, as $RBPBBased@N$, and the actual RBP score $b(r)$ of a run r, computed using a binary relevance:

$$b(r) = (1-p) \sum_{d \in r : j(d)=1} p^{\rho(d,r)-1} \quad (8)$$

The candidate documents for pooling are decreasingly ranked by:

$$M(d, R_p) = \sum_{r \in R_p, d \in r} RBPC(\rho(d, r)) =$$

$$= \sum_{r \in R_p : d \in r} (1-p)p^{\rho(d,r)-1} \cdot e(r) \cdot \left(b(r) + \frac{e(r)}{2} \right)^3 \quad (9)$$

Figure 1 shows the gain function variation with rank for the different pooling strategies, for one run r. The $RBPBBased@N$ and $RBPCBased@N$ strategies are not shown on this plot since, due to their adaptive nature, their shape changes with each judged document.

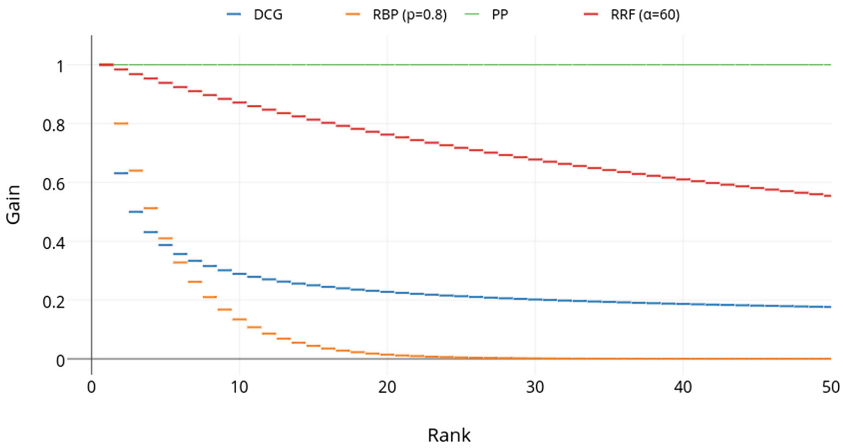

Fig. 1. Different pooling strategies score document ranks differently: This figure summarizes the gain functions in $DCGBased@N$, $RRFBased@N$, $PPBased@N$, and $RBPABased@N$ as functions of the rank position, for a run r.

3 Experiments and Results

The first part of this section describes the experimental set-up we have used. We list the test collections we made use of, the measures to assess the pool bias and the experimental methodology – similar to the one presented in previous studies [3,12–15]. The second part presents the results of the experiments with a series of plots.

3.1 Experimental Setup

To test the pooling strategies we used a set of 11 test collections selected from different editions of the TREC evaluation campaigns. We used Ad Hoc 2–5, Ad Hoc 7–8, Web 9, Web 10, Web 11, Genomics 14 and Robust 14. These test collections have been built using a *Depth*@k strategy, but additional documents have later been judged when additional resources were available. Therefore to remove the influence of these spurious assessments we preprocessed the test collections to use a pure *Depth*@k pool. The pool details for each resulting test collection are shown in Table 1.

To evaluate[1] the selected pooling strategies we ran experiments that simulated the absence of a run from the pool. We did this for every run, in a *leave-one run-out* fashion, then we summarized the bias with the following pool bias measures: Mean Absolute Error (MAE), System Rank Error (SRE), and System Rank Error with Statistical Significance (SRE*). MAE measures the mean of the error between the run score when the run contributed to the pool and its

Table 1. Pool properties of test collections, for the original pool and the *Depth*@100 (strategy D) pool; $|R|$ number of runs; $|R_p|$ number of pooled runs; $|O|$ number of organizations; $|T|$ number of topics; $|Q|$ number of judged documents; and $|Q_+|$ number of relevant documents.

Test Collection Properties												
	Ad Hoc 2		Ad Hoc 3		Ad Hoc 4		Ad Hoc 5		Ad Hoc 7		Ad Hoc 8	
$\|R\|$	38		40		33		61		103		129	
$\|R_p\|$	30		21		19		53		64		66	
$\|O\|$	22		22		19		21		42		41	
$\|T\|$	50		50		50		50		50		50	
	Orig.	→ D@100	Orig.	→ D@200	Orig.	→ D@100	Orig.	→ D@100	Orig.	→ D@100	Orig.	→ D@100
$\|Q\|$	62,620	39,692	97,319	68,121	87,069	46,721	133,681	71,448	80,345	69,662	86,830	79,090
$\|Q_+\|$	11,645	9,489	9,805	8,607	6,503	4,622	5,524	4,333	4,674	3,986	4,728	4,090

	Web 9		Web 2001		Web 2002		Robust 2005		Genomics 2005	
$\|R\|$	104		97		69		74		62	
$\|R_p\|$	39		35		60		18		46	
$\|O\|$	23		29		16		17		32	
$\|T\|$	50		50		50		50		49	
	Orig.	→ D@100	Orig.	→ D@100	Orig.	→ D@50	Orig.	→ D@55	Orig.	→ D@60
$\|Q\|$	70,070	49,161	70,400	46,135	56,650	53,318	37,798	22,173	39,958	32,013
$\|Q_+\|$	2,617	2,225	3,363	2,833	1,574	1,487	6,561	4,563	4,584	3,937

[1] The software used in this paper is available on the website of the first author.

score when left out. SRE is the sum of the rank error measured for each run, that is the difference in system ranking when the run contributed to the pool and when left out. SRE* is similar to SRE but counts the ranking difference only when statistical significance occurs (paired t-test $p < 0.05$).

To remove the influence that other contributing runs from the same organization may provide to the excluded run, we do instead a *leave-one organization-out*. We also remove the 25% of poorly performing runs, as done in previous studies [3,15]. To avoid also the discovery for each strategy of documents for which we do not know their relevance, that is they have not been judged in the original pool, we allow the selection of the documents to be pooled only from the top of the runs; we cut the runs at the depth k of the original $Depth@k$ used to build the original pool.

To analyse the performance of each strategy at different fixed-cost budgets we test each strategy, varing the number of documents required to be judged from 5,000 to the size of the original pool in steps of 5,000. We selected three IR evaluation measures because: (1) they are common evaluation measures used in IR and (2) they present properties that are common across the majority of IR evaluation measures: top-heaviness (relevant documents at the top of the list are given more weight), utility based, and strongly correlated to the number of relevant documents retrieved. The IR measures are: MAP, NDCG, and P@100.

3.2 Results

Figure 2 shows the results we obtained for the TREC-8 Ad Hoc test collection, where we observe how the different pooling strategies behave for various numbers of total documents judged. Figure 3 shows the same data as Fig. 2 from a different view: it shows the performance of the different pooling strategies compared to the $Take@N$ strategy. In Fig. 4 we show the performance for the NDCG measure on the other 10 test collections.

4 Discussion and Conclusion

In the paper we can distinguish two categories of strategies: (1) the non adaptive ones formed by RRF, PP, $RBPA$, and DCG, and (2) the adaptive ones formed by $RBPB$ and $RBPC$. Note that $RBPC$ not only uses information on whether a document is judged or not, but also concerning its relevance.

$RBPC$ is the best performing strategy in all the test collections over MAP and NDCG as evaluation measures, and across all pool bias measures. Nevertheless it is the most difficult to operationalise as the pool needs to be built on the fly, a concern expressed before in the work of Lipani et al. [15].

Based on Fig. 3, we can clearly identify two different types of behaviour depending on the IR evaluation measure used. On one hand, both MAP and NDCG have similar behaviour. For these evaluation measures, $RBPB$ and $RBPC$ are the best strategies, followed by $RBPA$, DCG, RRF and PP. These last four pooling strategies have a similar behaviour characterized by a twist

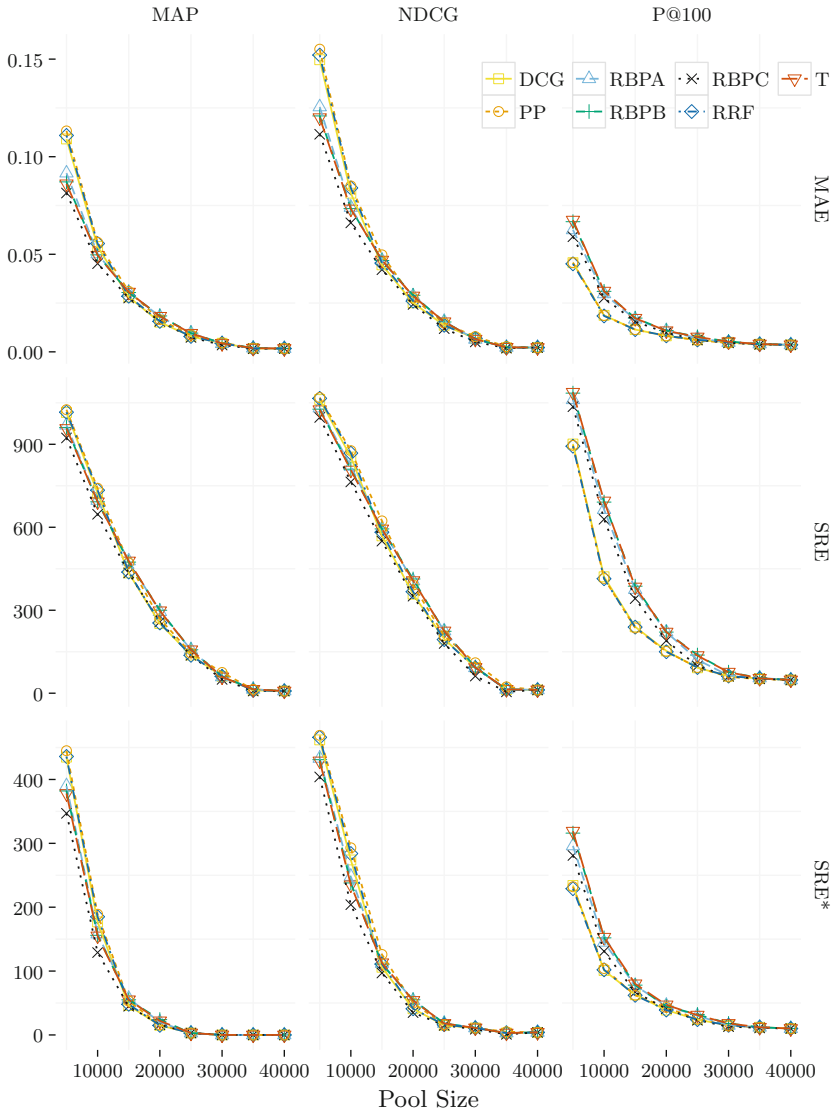

Fig. 2. Pool bias measured in terms of MAE, SRE, and SRE* for the pooling strategies on the Ad Hoc 8 test collection, for different pool sizes (i.e. number of documents that require relevance judgement).

between 10,000 and 15,000 judged documents in the case of the TREC-8 Ad Hoc collection. We also observe that a similar shape happens for the rest of the test collections, in Fig. 4 for NDCG.

The rank of the non-adaptive strategies is perfectly correlated with their speed of discount (change in reward for popularity) as observed in Fig. 1.

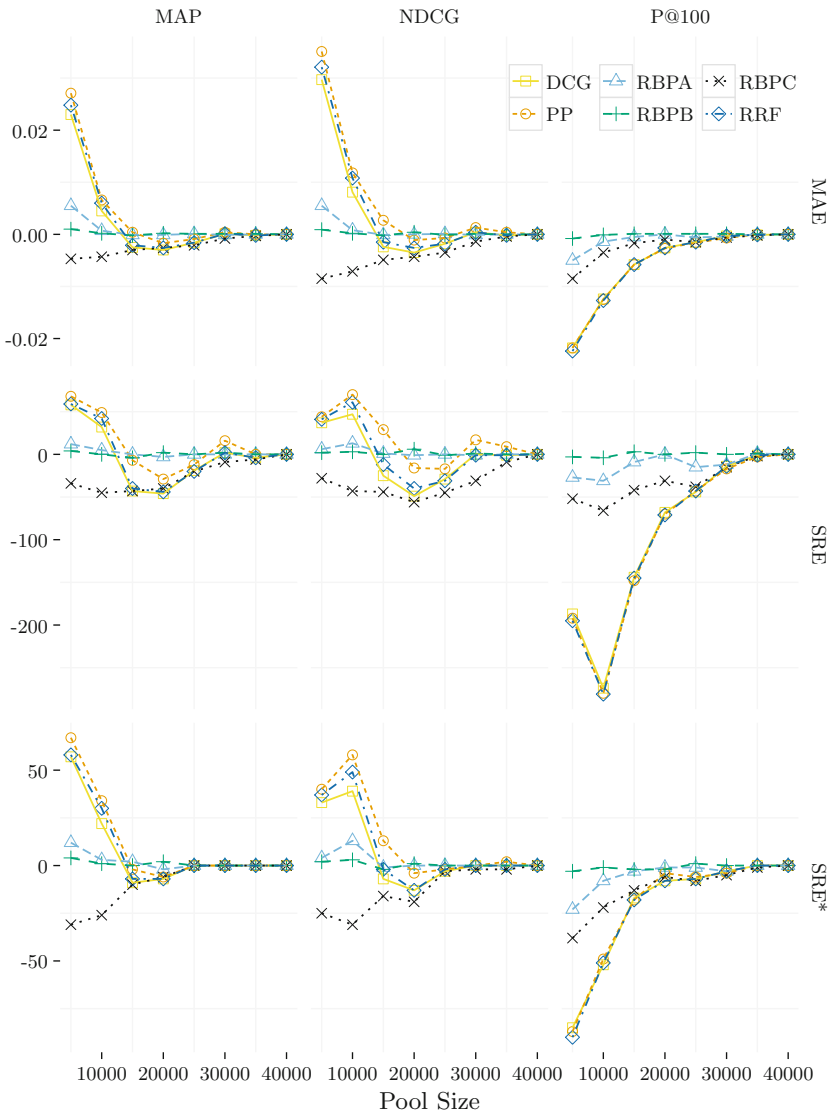

Fig. 3. Pool bias measured with respect to $Take@N$ strategy in terms of MAE, SRE, and SRE* for the pooling strategies on the Ad Hoc 8 test collection, for different pool sizes (i.e. number of documents that require relevance judgement).

The linear and logarithmic discounts remove the rank information from the documents rewarding more popularity of a document among the runs. The relationship between the discount and the top-heaviness of the evaluation measures MAP and NDCG also explains the twist in preference, where $Take@N$ is preferred for low N, then for higher N almost all non-adaptive methods outperform

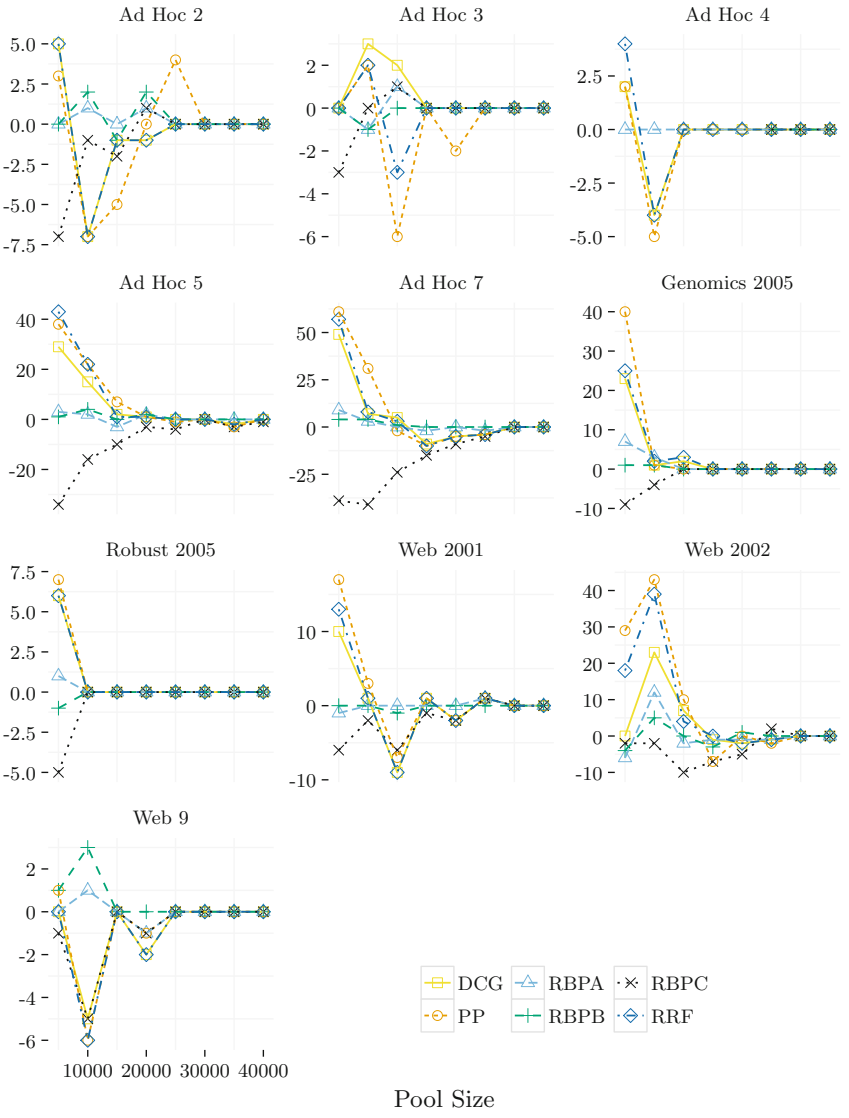

Fig. 4. Pool bias measured on NDCG with respect to $Take@N$ strategy in terms of SRE* for the pooling strategies on the rest of the test collections, for different pool sizes (i.e. number of documents that require relevance judgement).

it, before they all converge to the same value. On the other hand, for P@100 we observe that DCG, RRF, and PP are the best, followed by $RBPC$, $RBPA$, and $RBPB$. The latter behaves very similarly to the baseline $Take@N$. P@100 correlates with the number of relevant documents discovered in this specific case because the size of the submitted runs equals the original depth of the pool, and we justify its different behaviour due to the absence of discount.

It is strange here that $RBPA$ outperforms $RBPB$: a non-adaptive strategy outperforms an adaptive one. At this point we can only hypothesise that the exponential decay of $RBPA$ fights popularity rewarding more the rank.

In the end, the conclusions we can draw at this point are as follows:

- for top-heavy metrics:
  - given a large budget, the $Take@N$ strategy is guaranteed to be the least biased and therefore should be used;
  - given a small budget with which only very few documents can be assessed, then we should operationalise $RBPC$. It might take longer to create the assessments and it can only be done by one assessor per topic, but this would be likely in line with the budget constraints;
  - for a moderate budget and if we cannot operationalise $RBPC$, the non-adaptive strategies do not underperform $Take@N$, but neither do they consistently improve upon it.
- for $P@100$ it appears that the non-adaptive methods always outperform the baseline $Take@N$ and are therefore to be used. This is not only based on Ad Hoc 8 (Fig. 3), but is clearly visible for all test collections.

References

1. Aslam, J.A., Pavlu, V., Yilmaz, E.: A statistical method for system evaluation using incomplete judgments. In: Proceedings of SIGIR (2006)
2. Buckley, C., Dimmick, D., Soboroff, I., Voorhees, E.: Bias and the limits of pooling for large collections. Inf. Retr. **10**(6), 491–508 (2007)
3. Buckley, C., Voorhees, E.M.: Evaluating evaluation measure stability. In: Proceedings of SIGIR (2000)
4. Büttcher, S., Clarke, C.L.A., Yeung, P.C.K., Soboroff, I.: Reliable information retrieval evaluation with incomplete and biased judgments. In: Proceedings of SIGIR (2007)
5. Carterette, B.: System effectiveness, user models, and user utility: a conceptual framework for investigation. In: Proceedings of SIGIR (2011)
6. Cleverdon, C., Mills, J.: Factors determining the performance of indexing systems. In: Volume I - Design, Volume II - Test Results, ASLIB Cranfield Project (1966). (Reprinted in Sparck Jones, K., Willett, P. (eds.) Readings in Information Retrieval)
7. Cormack, G.V., Clarke, C.L.A., Buettcher, S.: Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In: Proceedings of SIGIR (2009)
8. Dumais, S., Banko, M., Brill, E., Lin, J., Ng, A.: Web question answering: is more always better? In: Proceedings of SIGIR (2002)
9. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of IR techniques. ACM Trans. Inf. Syst. **20**(4), 422–446 (2002)
10. Jones, K.S., van Rijsbergen, C.J.: Report on the need for and provision of an "ideal" information retrieval test collection. British Library Research and Development Report 5266, University of Cambridge (1975)
11. Lipani, A.: Fairness in information retrieval. In: Proceedings of SIGIR (2016)
12. Lipani, A., Lupu, M., Hanbury, A.: Splitting water: precision and anti-precision to reduce pool bias. In: Proceedings of SIGIR (2015)

13. Lipani, A., Lupu, M., Hanbury, A.: The curious incidence of bias corrections in the pool. In: Ferro, N., Crestani, F., Moens, M.-F., Mothe, J., Silvestri, F., Nunzio, G.M., Hauff, C., Silvello, G. (eds.) ECIR 2016. LNCS, vol. 9626, pp. 267–279. Springer, Cham (2016). doi:10.1007/978-3-319-30671-1_20
14. Lipani, A., Lupu, M., Palotti, J., Zuccon, G., Hanbury, A.: Fixed budget pooling strategies based on fusion methods. In: Proceedings of SAC (2017)
15. Lipani, A., Zuccon, G., Lupu, M., Koopman, B., Hanbury, A.: The impact of fixed-cost pooling strategies on test collection bias. In: Proceedings of ICTIR (2016)
16. Moffat, A., Webber, W., Zobel, J.: Strategic system comparisons via targeted relevance judgments. In: Proceedings of SIGIR (2007)
17. Moffat, A., Zobel, J.: Rank-biased precision for measurement of retrieval effectiveness. TOIS **27**(1), 2 (2008)
18. Park, L.A., Zhang, Y.: On the distribution of user persistence for rank-biased precision. In: Proceedings of ADCS (2007)
19. Robertson, S.: On the history of evaluation in IR. J. Inf. Sci. **34**(4), 439–456 (2008)
20. Sakai, T.: New performance metrics based on multigrade relevance: their application to question answering. In Proceedings of NTCIR (2004)
21. Soboroff, I.: A comparison of pooled and sampled relevance judgments. In: Proceedings of SIGIR (2007)
22. Voorhees, E.M., Harman, D.K.: TREC: Experiment and Evaluation in Information Retrieval. The MIT Press, Cambridge (2005)
23. Webber, W., Park, L.A.: Score adjustment for correction of pooling bias. In: Proceedings of SIGIR (2009)
24. Yilmaz, E., Aslam, J.A.: Estimating average precision with incomplete and imperfect judgments. In: Proceedings of CIKM (2006)
25. Zhang, Y., Park, L.A., Moffat, A.: Click-based evidence for decaying weight distributions in search effectiveness metrics. Inf. Retr. **13**(1), 46–69 (2010)
26. Zobel, J.: How reliable are the results of large-scale information retrieval experiments? In: Proceedings of SIGIR (1998)