The Solitude of Relevant Documents in the Pool

Aldo Lipani\textsuperscript{1} Mihai Lupu\textsuperscript{1} Evangelos Kanoulas\textsuperscript{1} Allan Hanbury\textsuperscript{1}
\textsuperscript{1}Inst. of Software Technology & Interactive Systems, Vienna University of Technology, Vienna, Austria \{lipani, lupu, hanbury\}@ifs.tuwien.ac.at
\textsuperscript{2}Informatics Institute, University of Amsterdam, Amsterdam, The Netherlands e.kanoulas@uva.nl

ABSTRACT

Pool bias is a well understood problem of test-collection based benchmarking in information retrieval. The pooling method itself is designed to identify all relevant documents. In practice, ‘all’ translates to ‘as many as possible given some budgetary constraints’ and the problem persists, albeit mitigated. Recently, methods to address this pool bias for previously created test collections have been proposed, for the evaluation measure precision at cut-off ($P@n$). Analyzing previous methods, we make the empirical observation that the distribution of the probability of providing new relevant documents to the pool, over the runs, is log-normal (when the pooling strategy is fixed depth at cut-off). We use this observation to calculate a prior probability of providing new relevant documents, which we then use in a pool bias estimator that improves upon previous estimates of precision at cut-off. Through extensive experimental results, covering 15 test collections, we show that the proposed bias correction method is the new state of the art, providing the closest estimates yet when compared to the original pool.

Keywords

Pool bias, $P@n$, test collections, TREC

1. INTRODUCTION

The pool bias refers to an undesirable side effect caused by the use of the pooling method. This bias manifests itself by the discounting of Information Retrieval (IR) systems when tested on previously built test collections, due to the retrieval of potentially relevant but non-judged documents [4].

An IR test collection is composed of: a collection of documents, a set of topics, and ideally a complete set of paired relations between topics and documents called relevance assessments, which indicate whether a document is either relevant or non-relevant for a given topic. Soon in the history of IR, with the explosion in size of the document collections and therefore of the number of relevance assessments required, the building of such test collections became impractical; to address this issue, the pooling method was introduced [9].

The main advantage of the pooling method is to limit the number of relevance assessments through the use of previously collected search results provided by various retrieval systems. The first introduced and most commonly used pooling strategy is the fixed-depth at $n$. This strategy consists in selecting the top $n$ documents retrieved by each retrieval system, then collecting them in a set, called the pool, which later will be fully judged. Although its main limitation is due the introduced incompleteness of the relevance assessments, as pointed out by Spärck Jones [3], the aim of the pooling method is not to find all the relevant documents but an unbiased sample of them. However, this is not an easy problem because, in the absence of full evaluation, and in the presence of millions of potentially relevant documents, there is no guarantee that a future retrieval method will not retrieve completely different yet relevant documents.

There are three ways to mitigate the pool bias: 1) increasing the depth of the pool, 2) increasing the number of topics, which would reduce the variance and, 3) increasing the number of pooled runs, which should increase the diversity of the pooled documents. Each of them has as effect an increment on the number of judgments to be performed. However, only the first two solutions are directly controllable by the test collection builder, leaving the third to the participation of the IR community in solving the challenge for which the test collection is going to be built.

To tackle the pool bias issue, the IR community has pursued two research paths, one developing pooling strategies aimed at mitigating the pool bias when creating new test collections [2, 8, 7] and the other, developing pool bias estimators aimed at correcting the observed pool bias on existing test collections. Witnessing an explosion in the number of test collections developed each year that use the pooling method with fixed-depth at $n$ pooling strategy and an increasing bias due to an increasing trend in the creation of test collections for niche domain-specific IR, where a required minimal participation is usually not met, in this paper we extend the work done on the latter research path. We develop a new pool bias estimator for the metric Precision at cut-off ($P@n$) based on empirical observations. There are two reasons for working with such a simple metric: there is an increasing demand from practitioners for metrics that make "sense" [1], and it is a corner stone for other more complex metrics.

2. BACKGROUND

Webber and Park [10] introduced a method to correct the pool bias, which they tested on Rank-Biased-Precision at cut-off but claimed to work also with $P@n$. This method adds to the score of a new run a coefficient equal to the mean difference, indicated with $\delta P@n$, between the score obtained when a run, initially part of the pool
$R_p$, is pooled and not-pooled:

$$\delta P\overline{\otimes n}(r_p) = P\otimes n(r_p, Q_{R_p}) - P\otimes n(r_p, Q_{R_p}\setminus\{r_p\})$$

(1)

The correction coefficient for a run ($r_u \notin R_p$) is the expectation:

$$E_{r_p \in R_p}[\delta P\overline{\otimes n}(r_p)]$$

(2)

where $Q_{R_p}$ is the set of judged documents created using the set of pooled runs $R_p$. This approach assumes that any given new run is sampled from the same distribution as the pooled ones. This is of course not always true, because runs are selected based on their performance by human intervention. A limitation of this approach is that it computes a coefficient that is constant and therefore does not depend on the actual status of the biased run. Another limitation of this approach is that the correction is not bounded by the tested number of this hypothesis, we observe in Fig. 1, on the top, that the distribution of the $\bar{\delta}P/\bar{\delta}n$ is normal. This assumption implies that runs are similar to each other, which is of course not true. To empirically demonstrate the groundlessness of this hypothesis, we observe in Fig. 1, on the top, that the distribution of the $\delta P/\delta n$ is not normally distributed. In the Q-Q plot in Fig. 1 on the bottom, we observe how the theoretical normal distribution correlates with the sample distribution. To calculate the mean of the new estimate, we compute the mean of the distribution $Y$ and then transform it back to the domain of the distribution $X$, which leads to the geometric mean of the $X$:

$$e^{E[\ln(X)\mid Y]} = e^{E[Y\mid Y]} = \sqrt[n]{\prod_{y \in Y} y} = GM[X]$$

(4)

where GM is the geometric mean. Thereby, the correction coeffi-
For a run \((r_u \not\in R_p)\) the following function:

\[
k@n(r_u) \cdot \text{GM}_{r
\in R_p} \left[ \delta \text{P} @ n(r_p) / k' @ n(r_p) \right]
\]

(5)

Comparing Eq. (2) and Eq. (5) we notice that the numerator of this last equation is the same, but with the difference that now, every \(\delta P @ n\) gets divided by the number of uniquely identified documents provided to the pool, and then multiplied by the same but for \(r_u\), and the arithmetic mean has been substituted by the geometric mean.

However, Eq. (4) is only valid when \(X\) is well behaved in the sense of not having zero values. Considering this, in Eq. (5) it mandates the removal of those runs for which \(\delta P @ n\) is zero. This is reasonable because: first, if \(\delta P @ n \neq 0\) it can be shown that \(k'_r \neq 0\) and consequently the fraction in Eq. (5) is well defined. Second, such zero values bring no information to our estimate of the contribution of the run. In fact, one could generate an unbounded number of runs with \(\delta P @ n = 0\).

In Alg. 1 we make use of Eq. (5), filtering out 0 values, and redefining the previous Webber approach. Two benefits of this modification are that no longer every correction is based on a priori probabilities of the run to find relevant documents among its non-judged ones, and that it is not a constant number as it was before. Also, comparing the algorithm of the Lipani approach [5], we observe here some similarities, such as the use of \(k @ n\) and of \(T @ n\).

4. EXPERIMENTS & RESULTS

In this section, we present the experimental setup and the material used. It follows the methodology previously used by the already mentioned studies [10, 5, 6]. In the second part, we discuss the results.

4.1 Material & Experimental Set up

To test the effectiveness of the new algorithm, we tested\(^1\) the method on 15 test collections sampled from the TREC evaluation campaign: 7 from the Ad Hoc track, 3 from the Web track, and 1 from each of the Genomics, Robust, Legal, Medical, and Microblog tracks. This sample allowed us to also explore more specific IR domains, where a different behavior may be expected. The experiments consist in running repeated simulations, where a run at each iteration is taken out of the pool with all the runs submitted by the same organization, and a synthetic pool is generated, in a leave-one-organization-out fashion. This is done to better simulate the absence of any similar run added to the pool by the same organization. Moreover, to avoid the presence of runs produced by buggy implementations of IR search engines, the bottom 25% of poorly performing runs are filtered out.

To measure the effectiveness of the different approaches, three measures are used: Mean Absolute Error (MAE), System Rank Error (SRE) and System Rank Error with statistical significance (SRE*). MAE measures the mean of the absolute difference of runs’ scores when in the pool and when not. SRE measures the difference in ranks between the true ranking and new ranking with using the corrections. Finally, SRE* is similar to SRE, but counts the difference only if statistically significant (paired t-test, \(p < 0.05\)). We compared the new approach with the previously mentioned Webber and Lipani approaches, and the baseline, which is the Reduced Pool. The Reduced Pool measures the bias we get if no correction is provided.

4.2 Results

In Tab. 1 the full results are presented. We observe that our approach is top performer in the majority of the cases (45 for MAE, 37 for SRE and 9 for SRE*), with some worst results (15 for MAE, 16 for SRE and, 6 for SRE*) mostly obtained on Ad Hoc 7 and Ad Hoc 8, where the Lipani approach performs better. The Lipani approach, although in general it is not as good as our approach, which has a more stable behavior, collecting fewer worst results (9 for MAE, 6 for SRE and, 1 for SRE*). The Webber approach and the Reduced Pool are the ones that get the majority of the worst results, as also demonstrated in previous work [5].

When our approach is applied to the test collections: Ad Hoc 3, Ad Hoc 5, Ad Hoc 7, Ad Hoc 8, Web 9, Web 2002, Genomics 2005 and Microblog 2011 we can observe that the MAE for the correction applied to \(P \oplus 5\) gets the worst score (8 of the 15 worst performer cases mentioned above). Here, after an empirical analysis, we have two hypotheses for why this happens, and we claim that it happens for a combination of both. The first hypothesis relies on the observation that for such test collections, the ratio between the number of pooled runs and the number of organizations is much greater than 1. This means that multiple runs from the same organization have been pooled and therefore contribute a very similar set of documents to the pool. Thereby, it nullifies the leave-one-run-out approach embedded in our and the Webber methods, as shown in Eq. (2) and (5), looking inside E in the first and GM in the second equation, when the run \(r_p\), originally in the pool \(Q @ n\) is removed from the pool \(Q @ n(1 @ r_p)\). Note that this leave-one-run-out approach is different from the leave-one-organization-out approach.

\(^1\)The software is available on the website of the first author.
of being or not being in the pool. Thereby, the error introduced is bigger than the one we would have observed if we had not corrected the run at all. In fact we can observe that the second best result, but for Genomics 2005, excluding the Lipani approach, is obtained by the Reduced Pool. Therefore to mitigate this effect, it is necessary to introduce a trigger that checks if the number of data points collected are sufficient to compute the estimate and perform the correction, and if not, to fall back to the Reduced Pool error.

5. CONCLUSION

We have presented new insights, empirically justified, to correct the pool bias for the evaluation measure P00m. The method is based on new insights on the fixed-depth at n pooling strategy, namely the observation of a relation between the probability of a run to provide uniquely identified relevant documents to the pool and the log-normal distribution. The proposed method largely outperformed the previous approaches, although for two test collections, Ad Hoc 7 and Ad Hoc 8, where the Lipani estimator works better. We identified two hypotheses that would explain why the reduced pool does not work for both of which one of their source to situations where, for the metric at hand, the vast majority of runs have all of their documents judged in the reduced pool. A cross comparison between the algorithm of the Lipani estimator and the algorithm presented here shows some similarities which may inspire a new improvement on the correction of the pool bias.

Acknowledgments

This research was partially supported by the Austrian Science Fund (FWF) project number P25905-N23 (ADMIRE), and by the Google Faculty Award Research Scheme. All authors represent the opinions of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

6. REFERENCES