Parameter Sensitivity in Rank-Biased Precision

Yuye Zhang *Laurence A. F. Park* *Alistair Moffat*

NICTA Victoria Laboratory
Department of Computer Science and Software Engineering
The University of Melbourne
Victoria 3010, Australia

{*zhangy, lapark, alistair*}*@csse.unimelb.edu.au*

Abstract *Rank-Biased Precision (RBP) is a retrieval evaluation metric that assigns an effectiveness score to a ranking by computing a geometricly weighted sum of document relevance values, with the monotonicly decreasing weights in the geometric distribution determined via a persistence parameter p. Despite exhibiting various advantageous traits over well known existing measures such as Average Precision, RBP has the drawback of requiring the designer of any experiment to choose a value for p. Here we present a method that allows retrieval systems evaluated using RBP with different p values to be compared. The proposed approach involves calculating two critical bounding relevance vectors for the original RBP score, and using those vectors to calculate the range of possible RBP scores for any other value of p. Those bounds may then be sufficient to allow the outright superiority of one system over the other to be established. In addition, the process can be modified to handle any RBP residuals associated with either of the two systems. We believe the adoption of the comparison process described in this paper will greatly aid the uptake of RBP in evaluation experiments.*

Keywords Rank-Biased Precision, Evaluation, System Comparison

1 Introduction

Effectiveness evaluation focuses on allocating scores to retrieval systems, allowing researchers to compare pairs of systems, and argue that one or the other has the better effectiveness. When using a non-parameterized metric, systems are simply compared by the effectiveness score computed for each system's set of retrieved relevance vectors. However, the task of comparing systems is complicated when adjustable effectiveness-scoring parameters are introduced, as it is difficult (or simply meaningless) to compare systems evaluated using different parameter values. This presents a problem for measures where there are no conventionally agreed values for those parameters, and results in a lack of any clear baselines being established as the basis for future improvements.

Unfortunately, many popular metrics such as NDCG [Järvelin and Kekäläinen, 2002] and BPref [Buckley and Voorhees, 2004] make use of such parameters, and it is often the case that the parameter needs to be adjusted to the experimental context or be chosen based on prior knowledge of the dataset properties. This problem also extends to metrics such as Average Precision, for which the depth of the evaluation is clearly a parameter that must be set by the experimental designer, and might play a large part in determining the numeric value of the scores that are achieved.

Rank-Biased Precision (RBP) [Moffat and Zobel, 2009] is an evaluation measure which assigns relevance weights based on the geometric distribution for a given parameter $0 \leq p < 1$ or *persistence*, where a smaller p value places greater emphasis on documents that appear early in the ranking, and a larger p spreads the weight further down the document ranking, but in both cases with all documents in the ranking contributing to the final score.

Despite the merits of RBP, there is no single "best" p that can be used for experimentation, as p is by its very nature something that is varied across different types of experiment. Here we present a method to compare RBP scores computed using differing p values, based on the bounding binary relevance vectors obtained by inverting the RBP score calculation.

Our methodology uses a three step process:

1. For a given p and RBP score, calculate the lexicographically greatest and least relevance vectors which might have generated that score (at some floating point precision).

2. Using these two vectors, calculate the range of possible RBP values for the target p value.

3. Finally, based on the target RBP value, we can deduce whether one RBP score is outright greater than the other.

Additionally, this method can be modified to work for RBP values with non-zero residuals, or known

Proceedings of the 13th Australasian Document Computing Symposium, Hobart, Australia, 8 December 2008.

imprecision. Our investigation into these properties shows that changing p results in interesting behaviors in RBP based on the source values, and that it is possible for outright comparisons to be performed on two RBP scores computed using different parameters. Our outcomes suggest that RBP can be employed more extensively in evaluation experiments than it is currently, with reduced concerns over incomparable results between researchers.

Section 2 introduces the RBP metric and other related work in the general research area. Section 3 describes the method to generate relevance vectors from an initial RBP score and associated p value, in particular the process to obtain the lexicographically greatest and least relevance vectors. Section 4 demonstrates how to obtain a range of RBP values using these two vectors and the interpretation needed to form a clear system comparison. Section 5 presents a modified process for handling the presence of RBP residuals. Section 6 examines some peculiarities and limitations of the comparison process. Finally, Section 7 concludes the paper and discusses possibilities for future investigation.

2 RBP and related metrics

Rank-Biased Precision (RBP) is based on the monotonically decreasing values in a geometric sequence. It has the form:

$$\text{RBP}(R, p) = (1 - p)\sum_{i=1}^{|R|} r_i p^{i-1}$$

where p is a abstraction of the user's searching persistence, expressed as a parameter between 0 and 1, R represents the relevance vector to be evaluated, and r_i indicates the relevance of the document ranked in position i within the ranking [Moffat and Zobel, 2009]. Unlike some other metrics, RBP does not utilize the global number of relevant documents for the query and is formulated in such a manner that a relevant document at any given rank contributes a set value to the overall score, meaning that potential contributions from unjudged documents can also be calculated and incorporated as they become available. Consequently, RBP is always bounded between zero and one, with a score of one only achievable when the length of an "every document is relevant" ranking vector approaches infinity.

As an example of how RBP is calculated, suppose that a user has persistence of $p = 0.5$, meaning that there is a 50:50 chance that the user will progress from one document in the ranking to the next. If a system returns the relevance vector $R = \{11010001\}$, where 1 denotes a relevant document and 0 denotes an irrelevant document, then the RBP of this system is computed as: $(1 - 0.5) \times (0.5^0 + 0.5^1 + 0.5^3 + 0.5^7) = 0.816$.

Figure 1 depicts the effect of three different values of p on the RBP contribution for a set of ranks. As p increases, the contribution from early ranked documents decreases, and later ranked documents increase their weight in the final RBP score. The specific properties of the geometric sequence imply that it is possible to determine the ranking depth required when evaluating to a given accuracy for any predetermined value of p. For example, when $p = 0.5$ the sum of contributions from rank 12 onwards is 4.88×10^{-4}. Therefore, when evaluating to a precision of three decimal places (0.001), it is possible to do so using relevance judgments up to just rank 11 (and no further), as 4.88×10^{-4} rounds to less than 0.001. We make use of this property later.

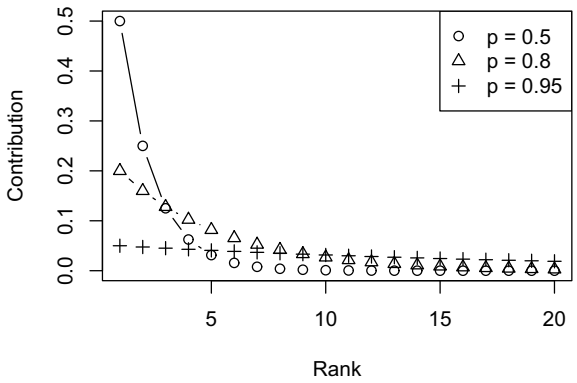

Figure 1: Contribution of each rank towards the RBP total for three values of p. Increasing the value of p shifts the emphasis to documents further down the ranked list.

Previous studies [Park and Zhang, 2007] suggest that for web search a p value of 0.8 is an appropriate value. In practice, values as high as 0.95 are used in experiments with higher pooling depths such as TREC [Voorhees and Harman, 2000], matching the deep evaluations provided by Mean Average Precision (MAP) and NDCG. Moffat and Zobel [2009] argue that this flexibility in choosing p works to RBP's advantage, allowing it to (mimic as required) the characteristics of other common evaluation metrics, such as Reciprocal rank, Precision@10, and so on.

Other recent studies have also examined RBP. Park and Zhang [2007] describe a method for selecting p based on session and click-through data from a large Microsoft query log. Moffat et al. [2007] investigate methods for reducing the number of relevance judgments required when performing system comparisons, and evaluated their methods in the context of RBP. Webber et al. [2008] describe a process for standardization using existing experimental results to modify evaluation metrics to shift reliance away from collection specific parameters. Compared to other evaluation metrics, Discounted Cumulative Gain (DCG) [Järvelin and Kekäläinen, 2002] bears many similarities to RBP, although overall DCG scores are unbounded as ranked lists grow longer, and the approach employed in Normalized Discounted Cumulative Gain (NDCG) requires prior knowledge of the relevance data. The BPref metric [Buckley and Voorhees, 2004] and the Q-measure metric [Sakai,

2004] are examples of more complex measures that make use of query-specific relevance information and the sequencing of relevant documents within the result list being evaluated.

Like all other metrics, RBP scores are easily comparable when the systems in question use the same parameter values, or supply the original rankings (runs), in which case scores using new parameters can be calculated. However, since not all experiments utilize identical parameter values, and published results typically only include summary data rather than details of the runs, alternative methods are required to compare systems evaluated using different p values.

3 Generating relevance vectors

As mentioned previously, it is possible to compare RBP scores computed using different p values as long as the original relevance vector is available. Our key contribution is the observation that it is possible to reconstruct a set of relevance vectors which could have given rise to any given RBP score.

Since the contribution of a relevant document at each rank is fixed for a given p, we can take a straightforward constraint-based approach to determine the values of ranks in the relevance vector. Given a retrieval system, with RBP score S obtained using persistence p, our goal of generating relevance vectors can be formally defined as calculating some set of relevance vectors $\mathcal{R}$, such that each relevance vector $R = \{r_1, r_2 \cdots\} \in \mathcal{R}$ satisfies the equation:

$$\text{RBP}(R, p) = S\,.$$

In our scenario of generating vectors R that satisfy S using p, we have no knowledge of any $r_i \in R$ and it is our goal to determine their values. We now define the following constraints:

Constraint 1 *Given R and p where r_j is determined for $0 < j < i$ and RBP$(R,p) < S$, if setting $r_i = 1$ causes the calculated RBP(R,p) to become greater than S, then either $r_i = 0$, or one of the earlier r_j values is incorrect.*

Constraint 2 *Given R and p where r_j is determined for $0 < j < i$ and RBP$(R,p) < S$, if setting $r_k = 1$ for $k > i$ still results in RBP$(R,p) < S$, then either $r_i = 1$, or one of the earlier r_j values is incorrect.*

An example demonstrates the use of the two constraints when deriving the required relevance vector. Suppose that the target score is $S = 0.4$, that $p = 0.8$, and that $R = \{1\ 0\ 0\ 0\ 1\ ?\ ?\ ?\}$, where ? represents a ranking with some unknown values. In this configuration, RBP$(R,p) = 0.2812$. If the options for r_6 are then considered, setting it to 1 does not break constraint 1, but setting $r_6 = 0$ breaks constraint 2, because the remaining unknown document judgments can only contribute at most 0.0943, which is not enough to allow the target of $S = 0.4$ to be reached (because $0.2812 + 0.0943 < 0.4$). Therefore $r_6 = 1$, giving $R = \{1\ 0\ 0\ 0\ 1\ \underline{1}\ ?\ ?\}$. The search can then continue.

	10^{-2}	10^{-4}	10^{-8}
$p = 0.5$	8	15	28
$p = 0.7$	15	28	54
$p = 0.8$	24	45	86
$p = 0.9$	51	94	182
$p = 0.95$	104	194	373
$p = 0.99$	528	986	1,001

Table 1: Number of significant ranks at given floating point precision as a function of p.

To formalize this process, let con(i) represent the contribution for a relevant document at rank i. Then, assuming binary relevance and some fixed p:

$$\text{con}(i) = p^{i-1} \times (1-p)\,,$$

and the contributions of all remaining documents from rank i onwards can be represented as:

$$\text{rem}(i) = \sum_{k=i+1}^{|R|} \text{con}(k)\ .$$

The constraints can now be expressed as:

$$r_i \in R = \begin{cases} 0 & \text{if acc}(i) + \text{con}(i) > S \\ 1 & \text{if acc}(i) + \text{rem}(i) < S \end{cases}$$

where:

$$\text{acc}(i) = \begin{cases} 0 & \text{if } i = 1 \\ \sum_{j=1}^{i-1} r_j \cdot \text{con}(j) & \text{otherwise} \end{cases}$$

In this approach, we evaluate the values of r_i sequentially by following the constraints, starting at r_1. Technically R can reach lengths up to infinity, but we bound this value by specifying a precision at which we stop the calculation. Hence, $|R|$ is simply the rank at which the rounded value of rem(i) becomes less than the precision. Table 1 depicts the number of results which are significant in terms of precision for various p and precision values.

For cases when neither constraint is satisfied, there is a choice. To obtain the full set of vectors $\mathcal{R}$, both possible values for r_i would be made at these *choicepoints*, and the search would then continue along both paths. However, we will take special note of two possible R vectors with useful properties: the one with the most relevant documents at the top of the ranking, obtained by always assigning $r_i = 1$ at choicepoints and denoted as R_G; and the one with the most irrelevant documents at the top of the ranking, obtained by always assigning $r_i = 0$ at choicepoints, denoted as R_L.

Both R_G and R_L are useful in that they depict the extremes of possible relevance combinations, being respectively the lexicographically greatest and lexicographically least. For our proposed method of RBP comparisons, simply making use of these two vectors is sufficient, meaning there is no need to generate all of $\mathcal{R}$.

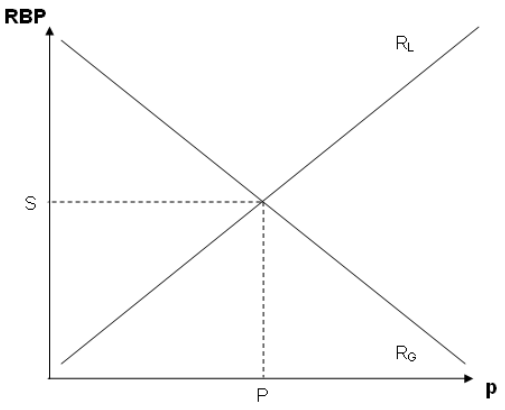

Figure 2: Changes in RBP value for R_G and R_L vectors for varying p values. As p increases, greater emphasis is placed on later ranked documents, pushing up the score assigned to the R_L vector, and decreasing the value assigned to the R_G vector. When p decreases, the converse is true. This figure illustrate a case in which R_L and R_G are divergent.

4 Comparing RBP values

The generated relevance vectors then allow calculation of the (range of) RBP scores that might have arisen if a different value of p had been used.

4.1 Bounding RBP scores for varying p

For a given initial p and score S, upper and lower limits on the RBP score can be computed for all other values of p, using the R_L and R_G vectors. These two values represent the extremes that can arise from any other members of $\mathcal{R}$, and yield the largest variations in RBP score as p is varied.

Recall that a floating point precision was specified to limit the length of the relevance vectors, with the implication that for a fixed precision and RBP score, a higher p' value (where $p' > p$) requires more ranks to be fully represented. This creates a undesirable situation where r_i of significant ranks at p' are unavailable, meaning we cannot use the R_G and R_L vectors in their current state as $|R|$ is too short to fully represent the RBP score at the required precision.

However, as we are primarily interested in the upper and lower bounds for possible RBP values, designating $r_i = 1$ with R_G and $r_i = 0$ with R_L for the extended rank positions in p' provides the greatest and least possible values for RBP respectively. When moving to a lower p value, the number of significant ranks decreases, meaning that $r_i = 0$ is appropriate for the extended ranks.

Figure 2 shows a representation of the range of possible RBP values obtainable from an initial p and S pairing. Intuitively, the range of possible RBP scores expands as p increases. All possible values in the range $[0, 1]$ eventually become obtainable as p asymptotically approaches 1. That is, as the effect of later ranked documents is accentuated, the score from the R_L vector increases, and the score from the R_G vector decreases.

On the other hand, when the value of p approaches 0, greater emphasis is placed on early ranked documents, until only the first ranked document is significant in the RBP computation. Indeed, below $p = 0.5$ the first document in the ranking dominates the sum of all of the other rank positions. This property allows us to easily predict the behavior of the R_G and R_L scores for smaller values of p: as p tends to zero, if $r_1 = 1$ in R_L, both scores converge to 1. Otherwise, if $r_1 = 0$ in R_G, both R_L and R_G scores will converge to 0. Furthermore, given the initial p and S, we can easily determine the value r_1, as $\text{con}(1) = (1-p) \times p^{1-1} = (1-p)$ and $\text{rem}(1) = 1 - \text{con}(1) = p$. Therefore, if $S \leq (1-p)$, r_1 must equal 0. Conversely, if $S \geq p$, then r_1 must equal 1. Figure 3 depicts this occurrence for nine different combinations of p and S. Note that one of the nine combinations in the matrix of possibilities is infeasible, and has not been plotted.

4.2 System comparison with RBP

We now have a process to handle our experimental scenario: suppose we have two retrieval systems (A & B) which executed the same query on identical datasets. The author of system A reported a RBP score of S_A calculated using p_A. Similarly, the author of system B reported a RBP score of S_B calculated using $p_B > p_A$. We need to determine, if possible, whether system A outperforms system B on that query or vice versa, using one or the other of the two values of p. Using the methods outlined above, we can accomplish this task by generating the R_G and R_L vectors of one system and comparing the range of possible RBP scores to those of the other at that system's p.

Although we have the choice of generating relevance vectors for either system (and thus attempting the comparison at either of the two values of p), it is prudent to use the system with the higher p value, and compare the range of RBP scores to the system with the lower p value. This is because moving from a higher p to a lower p places greater emphasis on contributions of early ranks, and later ranks are more likely to be uncalculated due to initial precision specification. The same effect was noted earlier when recalculating RBP for higher values of p.

With these considerations in mind, we now have a complete process for comparing systems evaluated using RBP with different p values:

1. For evaluation systems A and B, assuming $p_B > p_A$, generate R_G and R_L using p_B and S_B at the required level of accuracy.
2. For $p < p_B$, crop $|R|$ to the significant ranks at the given level of accuracy. Calculate $\text{RBP}(R_G, p)$ and $\text{RBP}(R_L, p)$.
3. For $p > p_B$, append $r_i = 1$ to R_G and $r_i = 0$ to R_L until the significant rank at the given level of accuracy is reached. Calculate $\text{RBP}(R_G, p)$ and $\text{RBP}(R_L, p)$.

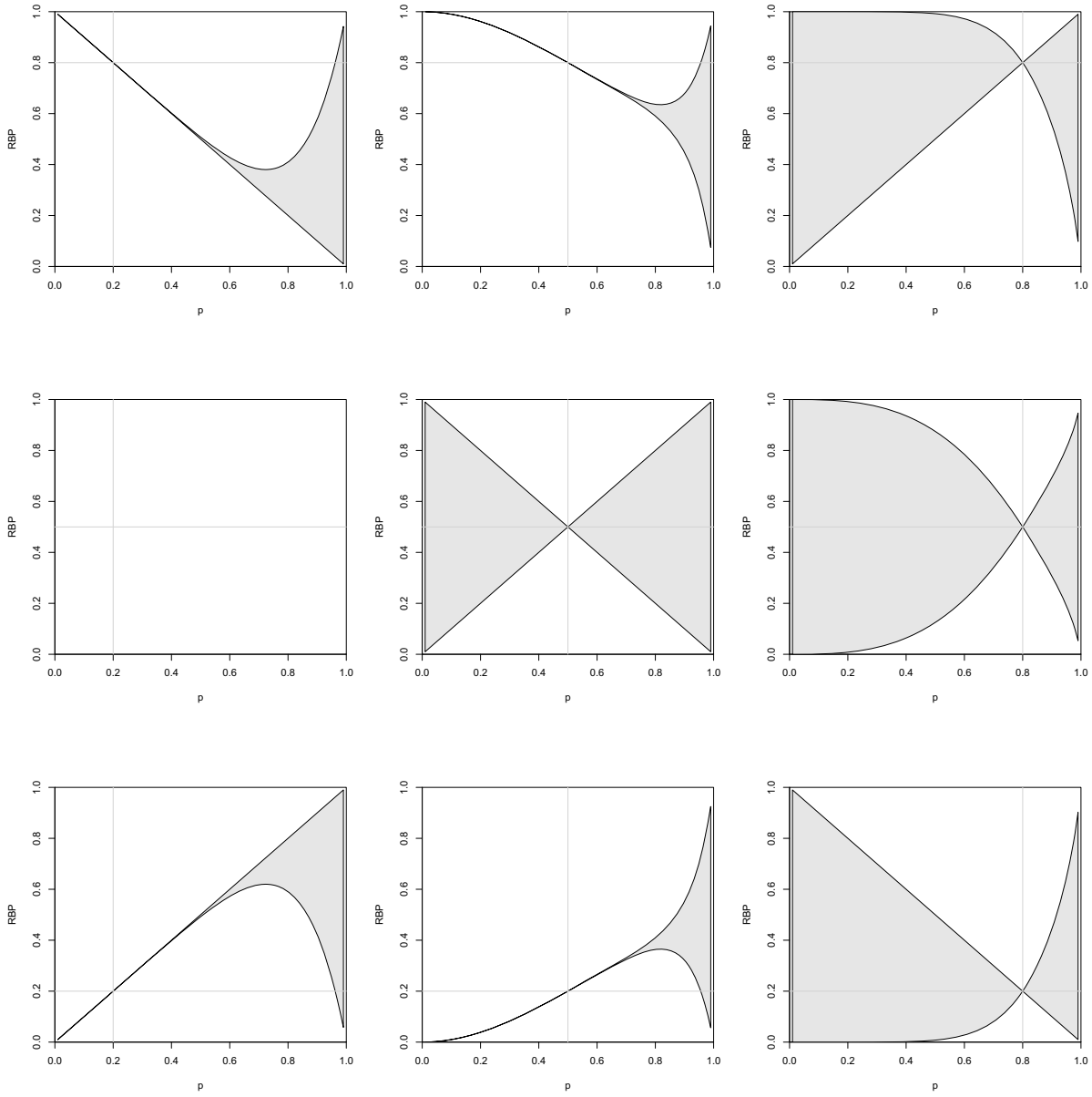

Figure 3: Convergence behavior using combinations p and S (RBP) values drawn from $\{0.2, 0.5, 0.8\}$. The bottom left corner shows $p = 0.2, S = 0.2$, and the top right corner shows $p = 0.8, S = 0.8$. Intersections occur at (p, S) in each graph, corresponding to the supplied arguments for generating R_G and R_L. The full divergent range of RBP scores in $[0, 1]$ becomes obtainable for values of p tending to 1, when the unspecified tail of the ranking has the power to completely change the score. Convergence of R_G and R_L for small values of p can be easily validated with be comparing p and S: both will converge to 1 if $S \geq p$ or both will converge to zero if $S \leq (1 - p)$. The graph of $p = 0.2, S = 0.5$ is missing because a score of 0.5 cannot arise when $p = 0.2$, as all valid RBP scores must be either greater than 0.8, or less than 0.2. Note that all of these bounds are based solely on the S and p values. If the actual ranking is available, RBP with a reduced p can always be calculated to at least the same accuracy as it was using the initial value of p.

4. System B outperforms system A at $p = p_A$ if and only if $S_A < \text{RBP}\,(R_L, p_A)$.

5. System A outperforms system B at p_A if and only if $S_A > \text{RBP}\,(R_G, p_A)$.

6. Otherwise, there is no clear outcome as to which system is superior.

The first two outcomes are fairly straightforward: if the RBP score of system A fails to reach lowest possible RBP score of system B, then system A is inferior. The second outcome is simply the mirrored case. However, when the RBP of system A lies in the range of system B, there is no clear evidence of superiority either way: out of all possible relevance vectors generated from system B (which fall between the RBP bounds marked out by system B's R_G and R_L), a non-empty subset of those vectors results in a higher RBP score at p_A, while others result in a lower score.

5 Integrating RBP residuals

Although the proposed method is generally applicable, retrieval experiments are often run with limited relevance judgments due to resource constraints, meaning that evaluation with exhaustive relevance judgments is impossible. In the case of RBP, this uncertainty is handled by summing the contributions of all unjudged documents to form an error bound, or *residual* [Moffat and Zobel, 2009]. The RBP residual (ε) can be described as:

$$\varepsilon(R, p) = (1 - p) \sum_{i=1}^{\infty} \text{unjudged}(i) \cdot p^{i-1}$$

where $\text{unjudged}(i) = 1$ if and only if r_i is unknown.

In this sense, the RBP score S of a ranking is the lower bound of RBP (if all unjudged documents are irrelevant), and $S + \varepsilon$ gives the upper bound, achieved if all unjudged documents are relevant. Ideally, both S and ε, along with the p employed, are reported when effectiveness evaluation results are being disseminated.

5.1 Relevance vectors with residuals

Residuals present a challenge when reconstructing the relevance vectors used, in that it is no longer valid to select r_i freely when choicepoints are encountered. This is because unlike previously discussed, the generated relevance vector R must be able to produce $S+\varepsilon$ should some subset of $r_i = 0$ positions be switched to $r_i = 1$, but still give S if they all remain unaltered. These modifications apply to both the R_G and R_L vectors.

Fortunately, we are able to incorporate these conditions into the original calculation process. We still need to abide by the original constraints so that the relevance vectors sums to the required S, but also have to integrate additional rules such that R_G and R_L is capable of satisfying $S + \varepsilon$ which certain positions are altered. Therefore, we integrate some new rules that affect the selection of r_i when we encounter a choicepoint.

For the R_G vector, we want to set $r_i = 1$ at choicepoints as long as the $r_i = 0$ positions (decided or otherwise) can contribute the equivalent of the residual. Furthermore, we have to take into account for ranks that haven't been decided, some subset of r_i that must be allocated to fulfilling the lower bound S:

$$r_i \in R_G = \begin{cases} 0 & \text{if } \text{acc}'(i) + \text{rem}\,(i) < \\ & \qquad S - (\text{con}\,(i) + \text{acc}\,(i)) + \varepsilon \\ 1 & \text{otherwise}\,, \end{cases}$$

where:

$$\text{acc}'(i) = \begin{cases} 0 & \text{if } i = 1 \\ \sum_{j=1}^{i-1} (1 - r_j) \cdot \text{con}\,(j) & \text{otherwise}\,. \end{cases}$$

For the R_L vector, we want to set $r_i = 0$ at choicepoints as long as the remaining contributions can reach the upper bound $S + \varepsilon$:

$$r_i \in R_L = \begin{cases} 1 & \text{if } \text{acc}\,(i) + \text{rem}\,(i) < S + \varepsilon \\ 0 & \text{otherwise}\,. \end{cases}$$

After integration of these rules, both R_G and R_L fit the criteria which allows some subset of their $r_i = 0$ judgments to be altered to reach the upper bound of the RBP score.

5.2 Positions of unjudged ranks

To supplement R_G and R_L, we determine the positions at which unjudged documents may occur in these vectors, taking

$$r_i = \begin{cases} \text{NA} & \text{if } \text{acc}\,(i) + \text{rem}'(i) < \varepsilon \\ 0 & \text{otherwise}\,, \end{cases}$$

where:

$$\text{rem}'(i) = \begin{cases} 0 & \text{if } i = 1 \\ \sum_{k=i+1}^{d} (1 - r_k) \cdot \text{con}\,(k) & \text{otherwise}\,. \end{cases}$$

Note that this calculation is only applied on ranks where $r_i = 0$, since we cannot change positions which have been previously decided to be relevant. Choicepoints can also be handled in a similar manner to determine the possible combinations of unknown judgment ranks, meaning we can consistently select $r_i = \text{NA}$ to shift unjudged documents towards the earlier ranks, or $r_i = 0$ to shift them to later ones.

Using the R_G and R_L vectors as a base, we now have four possible vector combinations. However, because we are most interested in the extremes of the relevance combinations, we let R'_G represent the lexicographically greatest vector with unknown judgments shifted towards the top of the ranking, and R'_L represent the lexicographically least vector with unknown judgments shifted towards the bottom of the ranking.

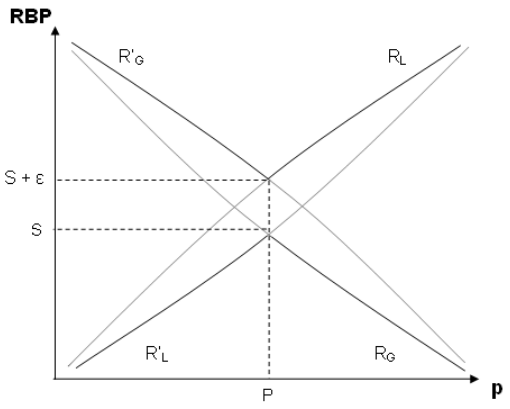

Figure 4: Change in RBP values for relevance vectors when p is varied. The darker lines indicates the upper and lower bounds for possible RBP values at any given p. This stylized figure illustrates the change in RBP bounds; and actual RBP bounds will vary.

5.3 System comparison with residuals

Using our previous definitions of S_A, p_A, S_B and $p_B > p_A$, we now introduce ε_A and ε_B to represent the RBP residual reported by each system. Following the convention of generating relevance vectors for system B as it has the higher p value, we now have bounds of $\{S_B, S_B + \varepsilon_B\}$ as the set of possible RBP values at p_B.

Firstly, we will deal with range of RBP scores possible at the lower bound of the initial RBP score. In this case, r_i = NA ranks in both R'_G and R'_L must be 0 for the score of S_B to be obtained. This implies the positions of the unjudged documents are not of importance, as only $r_i = 0$ positions were altered in creating R'_G and R'_L. We can simply calculated the possible RBP scores using the unaltered R_G and R_L vectors.

At the upper bound of the initial RBP score, all unjudged ranks in both R'_G and R'_L must be changed to 1 for the score of $S_B + \varepsilon_B$ to be obtained. We must now set r_i = NA to $r_i = 1$ in both vectors, and plot these in a similar manner. Figure 4 depicts the RBP bounds for a non-zero RBP residual.

We can see that in the case of a RBP residual being present, the upper and lower bounds on the possible RBP values are dictated by the magnitude of the residual at the initial p values. For larger values of p, the range of possible RBP values is dictated by R'_L and R_G, while for smaller values of p it is dictated by R_L and R'_G. Using these observations, our outcomes for the experimental scenario can be updated as follows:

1. System B outperforms System A if and only if $(S_A + \varepsilon_A) < \text{RBP}\,(R'_L, p_A)$.
2. System A outperforms System B if and only if $S_A > \text{RBP}\,(R'_G, p_A)$.

The interpretations for these outcomes are similar to the situation when no residual is present: one system outperforms the other only it's lowest possible RBP score at the given p is higher than the highest possible RBP score for the other system. In the case of overlap, it is still impossible to determine whether one system is better due insufficient knowledge about the generated relevance vectors.

6 Discussion

Although our method for comparing the RBP scores of different retrieval systems is relatively straightforward in terms of the processes involved, there remains a number of inherent characteristics (some intrinsic to the design of RBP itself) which should be taken into consideration.

Firstly, the initial process of generating relevance vectors is applicable for all values of p, although the number of possible relevance vectors generated varies:

- For $p = 0.5$, for all ranks $\text{con}\,(i) = \text{rem}\,(i)$ meaning there will always be exactly two vectors generated for all RBP values with either recurring 0 or 1 at the tail. These are the R_G and R_L vectors respectively. The range of possible RBP values as p is shifted is $[0, 1]$, as shown in Figure 3.
- For $p < 0.5$, some values of RBP are impossible to obtain: consider the case when $p = 0.2$ and $S = 0.5$. In this case, $\text{con}\,(1) = (1 - 0.2) \times (0.2)^0 = 0.8$ and $\text{rem}\,(1) = 0.2$ (assuming $d = \infty$), meaning it is impossible to obtain any RBP score between 0.2 and 0.8, as r_i cannot be assigned in any manner. In all other cases, there is a single unique vector. The range of possible RBP values is $[0, p]$ and $[1 - p, 1]$, with similar (and so on recursively) gaps within these two ranges.
- For $p > 0.5$, all possible values of RBP are obtainable. At a given level of accuracy, higher values of p will generate more potential vectors in $\mathcal{R}$ as there is a smaller variation in $\text{con}\,(i)$. The range of possible RBP values is $[0, 1]$.

Furthermore, because the generation process is based around determining potential contributions of individual ranks in the output vector, RBP scores obtained using non-binary relevance judgments are incompatible. This is because the additional variable introduced by the scaled judgments further confounds the range of choicepoints, and instead of seeking binary representations using a fractional radix, we are seeking n-ary ones.

Earlier we mentioned that when ranges of possible RBP values overlap, no unambiguous conclusion can be made in terms of relative superiority of systems. However, in the case when no residual is present in either system, supposing we have generated all possible relevance vectors $\mathcal{R}$ (instead of just R_G and R_L), calculating their values at the target p value will give a discrete set of possible RBP values instead of a range. Using basic probability measures it is tempting to determine a crude percentage chance that either system is superior.

Unfortunately, the shortcomings of this approach outweigh the benefits. The number of calculations required increases greatly for larger initial values of p, and the presence of RBP residuals in either system exacerbates the problem. The computation may be manageable for small residuals, but the overall expenditure does not justify the (arguably) limited usefulness of the information obtained. A simpler approach would be to calculate the numeric overlap of the two RBP regions, although this conveys even less information as $\mathcal{R}$ can vary greatly with different values of p, meaning the probability estimate will be quite likely to be skewed in some manner.

Finally, despite the fact that it is possible to calculate the range of RBP values at all values of p using the R_G and R_L vectors, the knowledge of upper and lower bounds of RBP for p greater than the original has limited usefulness. Although it was included in our illustrations for completeness, the upper and lower bounds represent extreme cases when the trailing documents are either all completely relevant or completely irrelevant. As such, these bounds should be used as guidelines only, rather than an indication that the comparison is possible.

7 Conclusion

We have presented a method comparing two systems evaluated using the Rank-Biased Precision effectiveness measure with different values for parameter p. This is achieved by generating the lexicographically greatest and least relevance vectors which can give rise to the original RBP score at the specified p, and using those two vectors to model the upper and lower bounds of possible RBP values at all other values of p. Furthermore, the generation and modeling process can be modified to handle RBP residuals, which will almost certainly be present in real world evaluation experiments.

By utilizing the processes we outlined, it may be possible for direct conclusions to be drawn regarding the superiority of one system over another, even though they have been scored using different values of p. This is a significant improvement from the current situation where there is no process to compare systems evaluated using varying values of p, and may aid in the uptake of RBP as a standard experimental metric. It is also possible that, with appropriate amendment, our proposed method for system comparison can be applied to other evaluation metrics that have fixed relevance contributions for each given position in the document ranking.

In terms of possible improvements, currently our method for performing system comparisons fails to deliver a clear outcome when there is an overlap between the RBP bounds of both systems at the required p value. It is not clear how this situation can be handled due to the limited information available when generating relevance vectors, and finding better approaches to this problem is a topic currently under investigation.

Finally, to get a sense of how often RBP bounds do in fact overlap when performing comparisons, it may be possible to utilize runs from existing TREC datasets and recalculate RBP scores using different p values. We can then compare different systems and observe how often one system is outright superior to the other, and establish a general idea of the applicability for our comparison method in its current form.

Acknowledgements This work was supported by the Australian Research Council. National ICT Australia (NICTA) is funded by the Australian Government's Backing Australia's Ability initiative, in part through the Australian Research Council.

References

Chris Buckley and Ellen M. Voorhees. Retrieval evaluation with incomplete information. In *Proc. 27th Ann. Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 25–32, Sheffield, United Kingdom, 2004. ACM Press, New York.

Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

Alistair Moffat and Justin Zobel. Rank-biased precision for measurement of retrieval effectiveness. *ACM Transactions on Information Systems*, January 2009. To appear.

Alistair Moffat, William Webber, and Justin Zobel. Strategic system comparisons via targeted relevance judgments. In *Proc. 30th Ann. Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 375–382, Amsterdam, The Netherlands, 2007. ACM Press, New York.

Laurence A. F. Park and Yuye Zhang. On the distribution of user persistence for rank-biased precision. In *Proc. 12th Australasian Document Computing Symp.*, pages 17–24. School of Computer Science and Information Technology, RMIT University, Australia, December 2007.

Tetsuya Sakai. Ranking the NTCIR systems based on multi-grade relevance. In *Proc. Asian Information Retrieval Symp.*, volume 3411, LNCS, pages 251–262. Springer, Berlin/Heidelberg, October 2004.

Ellen M. Voorhees and Donna Harman. Overview of the Ninth Text REtrieval conference (TREC-9). In *Proc. 2000 TREC Text Retrieval Conf.* National Institute of Standards and Technology, November 2000. `http://trec.nist.gov/pubs/trec9/papers/overview_9.pdf`.

William Webber, Alistair Moffat, and Justin Zobel. Score standardization for inter-collection comparison of retrieval systems. In *Proc. 31st Ann. Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 51–58, Singapore, July 2008. ACM Press, New York.