

Entropy-Based Query Performance Prediction for Neural Information Retrieval Systems

Oleg Zendel¹, Binsheng Liu², J. Shane Culpepper¹ and Falk Scholer¹

¹RMIT University, Melbourne, Australia

²SEEK, Melbourne, Australia

Abstract

Performance prediction is an important aspect of Information Retrieval (IR), as determining the effectiveness of search results without human relevance judgments has many important applications. We propose a novel Query Performance Prediction (QPP) method to predict the effectiveness of neural reranking models. Our approach uses the retrieval score distribution for a query and a set of highest-scoring documents to estimate the likelihood of effectiveness. This method is both efficient and unsupervised, making it possible to use in production retrieval systems. The new method uses entropy, which is the key measure in information theory. The core idea is simple but novel – measure the entropy of the retrieval scores for a reranking model while using no training data or corpus related statistics. Our empirical experiments show the effectiveness of our proposed method, which is comparable with traditional state-of-the-art QPP methods in terms of both prediction quality and computational efficiency.

Keywords

query performance prediction, neural information retrieval, information theory

1. Introduction

Query Performance Prediction (QPP) has been an important open problem in Information Retrieval (IR) research for many years [1, 2, 3, 4, 5]. Various methods have been proposed to predict the performance of a search result without relying on user interaction or human relevance judgments. Existing QPP methods are often divided into two categories: pre-retrieval and post-retrieval predictors. Pre-retrieval methods are generally retrieval system agnostic, but require access to various corpus statistics, and their prediction quality is usually lower than post-retrieval methods. Conversely, post-retrieval methods use a set of initial ranked results produced by a specific retrieval system. These methods are often computationally expensive [1, 6], and must be tuned for specific retrieval system [7].

More recently, neural Language Models (LMs) and neural re-ranking models have become increasingly popular in modern IR systems. However, Datta et al. [8] and Faggioli et al. [9], demonstrated that existing QPP methods are less effective for performance prediction when using neural ranking models. In this work, we propose a novel QPP method which leverages information induced from a neural reranking Retrieval Status Values (RSVs) to estimate the

QPP++ 2023: Query Performance Prediction and Its Evaluation in New Tasks, co-located with The 45th European Conference on Information Retrieval (ECIR) April 2, 2023, Dublin, Ireland


EMAIL: oleg.zendel@student.rmit.edu.au (O. Zendel)

ORCID: 0000-0003-1535-0989 (O. Zendel); 0000-0002-4084-4609 (B. Liu); 0000-0002-1902-9087 (J.S. Culpepper);

0000-0001-9094-0810 (F. Scholer)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

likelihood of effectiveness. In previous work, other properties such as the Standard Deviation (SD), mean, and magnitude of the RSVs have been shown to be correlated with retrieval effectiveness measures such as Average Precision (AP) [6, 7, 10]. In particular, SD, which is a standard measure of statistical dispersion, has received attention in prior work (for example see [7, 11, 12, 13, 14]).

Our work is based on the hypothesis that score distributions differ for highly and poorly ranked search results. In contrast to prior work on score dispersion, which focused on finding the number of top documents required to compute SD, we propose the use of entropy to capture the dispersion.

2. Related Work

A common type of post-retrieval QPP methods is score-based predictors. Score-based predictors use the retrieval scores (RSVs) generated by the retrieval system to estimate the success of the search. The simplicity and low computational cost of these methods make them an appealing choice in real-world applications as they do not require access to collection statistics, which can be computationally expensive. Additionally, they can be easily integrated into existing retrieval systems, and their predictions can be made in real-time.

A number of prior QPP methods estimate retrieval result effectiveness by measuring the dispersion of the RSVs with SD [11, 12, 13, 14]. One of the most widely used approaches is the Normalized Query Commitment (NQC) [7] method, which measures the SD of the retrieval scores, and includes additional corpus and query length normalization. One hypothesis in this work is that low dispersion of RSVs is likely to be evidence of query drift (the presence and dominance of documents not relevant to the information need). While SD based QPP methods have shown promising results, they are sensitive to the number of documents used in their estimation. This hypothesis inspires our approach – using entropy as a measure of dispersion and centrality. Our empirical analysis show that unlike SD, entropy is not sensitive to the number of documents being returned, and consistently improves the result as the number of top- k documents included increases.

Zendel et al. [15] propose a framework which includes additional reference queries in order to improve the estimate of the retrieval result effectiveness, using a linear combination of reference queries. This approach is extended by [8] to use a ratio instead, yielding better results for neural re-ranking models, and further improving the framework by generating query variants automatically. Both approaches enhance QPP estimators by incorporating information from additional reference queries, and depend on existing QPP predictors. Our new method can be used as a base predictor within these frameworks to further improve the effectiveness prediction of neural reranking models.

3. Experimental Setup

Retrieval methods. We test our proposed QPP approach using two probabilistic retrieval models: Query Likelihood (QL) [16]¹ and *NeuralRanker*. Both models estimate the probability

¹Using the Lemur-Indri toolkit.

Table 1

Average retrieval effectiveness scores measured with AP.

	QL	NeuralRanker
Robust04	0.248	0.284

$P(d|q)$ of document d being relevant to query q . *NeuralRanker*, based on the method originally described by Liu et al. [17], is trained by fine-tuning a Transformer-based LM on the MSMARCO passage ranking task, which has been shown to generalize well to other tasks and datasets – including document reranking [18]. At inference time, the model uses query-document pairs as input, and yields a relevance ranking score.

Evaluation methods. We report experimental results on an ad hoc retrieval TREC collection, the Robust04 [19] documents retrieval collection. We have tested our approach using other common collections, but due to space limitations, we do not include them here. The results were similar to the results shown for Robust04. Retrieval effectiveness is measured using AP, which is consistent with prior work on QPP. The AP values from QL and *NeuralRanker* are shown in Table 1.

To evaluate our approach, we employ both correlation measures, as was reported in prior work, as well as the recently proposed sARE and sMARE measures [20, 21]. Correlation measures used are Pearson’s r and Kendall’s τ . For hyperparameter tuning, we use repeated two-fold cross-validation with 30 repetitions. For each repetition, we report the evaluation measure as the average over the folds, and the final reported value is the average for all 30 repetitions.

Score-based QPP. Given the extensive and successful use of SD as a QPP method, and given that it is the method most similar to ours, we adopt it as the baseline for our comparisons. Our method is based on the entropy of the scores, normalized by softmax. Figure 1 illustrates the score differences observed for both good and bad queries. Each boxplot in the figure represents the distribution of the RSVs for the five best (worst) queries, measured by AP, on the Robust04 collection using *NeuralRanker*. Additional measures of dispersion, such as kurtosis, may also be useful; we leave such exploration for future work.

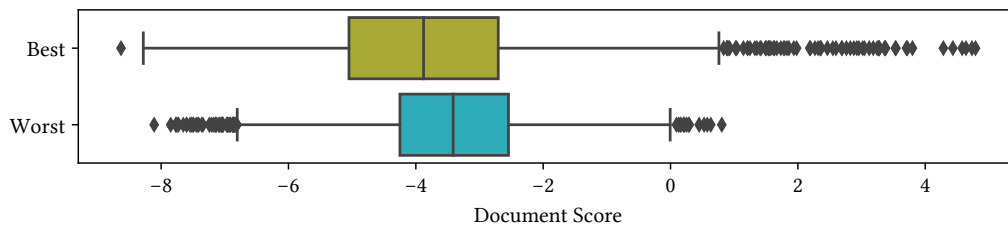
**Figure 1:** Boxplots of the RSVs distributions for best and worst queries.

Table 2

Prediction quality for SD and *entropy* as QPP measures. The more effective approach for each measure (column) is highlighted in bold. Note that sMARE is an error measure, hence lower is better. A * indicates a statistically significant difference between *entropy* and SD, as determined by Tukey’s Honestly Significant Difference (HSD) test with a Family-wise Error Rate (FWER) of 0.05.

		QL			NeuralRanker		
		sMARE	P-r	K- τ	sMARE	P-r	K- τ
Robust04	SD	0.224	0.481	0.345	0.220	0.526	0.359
	Entropy	0.221	0.385*	0.349	0.214*	0.529	0.388*

Entropy for QPP. The entropy of a query q is calculated based on the relevance probability distribution of a set of documents D_q :

$$H(q) = - \sum_{d \in D_q} P(d|q) \log P(d|q) \quad (1)$$

where $P(d|q)$ is the normalized probability of document d being relevant to the query q . The raw scores of QL and neural rankers, represented by $S(q, d)$, must be normalized to be interpreted as probabilities. The normalization is achieved using the *softmax* function:

$$P(d|q) = \frac{\exp(S(q, d))}{\sum_{d' \in D_q} \exp(S(q, d'))} \quad (2)$$

Entropy is maximized when the probabilities are uniform and indicates that the retrieval model was unable to effectively discriminate between good and bad documents. Conversely, lower entropy implies a more successful retrieval result. For convenience, in our experiments we use negative entropy, $-H(q)$, as our QPP method.

4. Experimental Results

A comprehensive evaluation on a benchmark dataset compares our method to the baseline SD method in terms of prediction quality, robustness, and similarity. Table 2 presents the prediction quality for both retrieval methods. It can be seen that for the task of document retrieval with the Robust04 collection, *entropy* performed comparably to SD, and was superior for the *NeuralRanker*. In terms of prediction quality, both methods produced similar results with only minor performance differences.

Hyperparameter analysis. Both the SD- and entropy-based QPP approaches make use of a hyperparameter, k , which represents the number of top-ranked documents that are included in the prediction calculation. Figure 2 shows the performance of the *entropy* and SD approaches, applied to predict the *NeuralRanker* results, with prediction quality measured using sMARE. It is clear that *entropy* is much more robust to variations in k than SD. The prediction quality of *entropy* remains consistent once it reaches $k \approx 400$, after which it shows diminishing returns.

SD, on the other hand, reaches the minimal error using a much narrower range of documents, which varies between collections and retrieval methods [7].

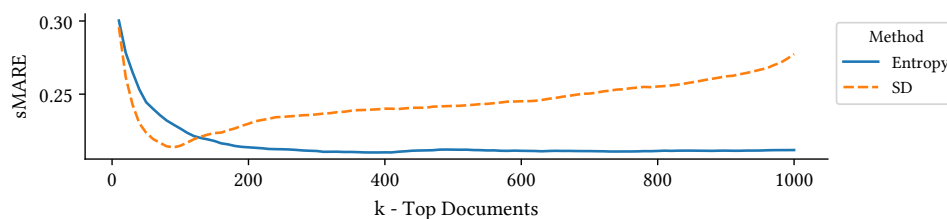


Figure 2: A comparison between *entropy* and *SD* as QPP methods, over a range of k values [10, 1000] with steps of 10, representing the number of top-ranked document scores used by the method. Prediction quality is measured with sMARE (lower is better) for NeuralRanker and AP.

Similarity of the predictions. In Figure 3, we evaluate both methods for the NeuralRanker results on the Robust04 collection by calculating the sARE (error) values per query, plotting them as a categorical scatter plot. The methods have a correlation of Pearson’s $r=0.5$. It can be seen that while some queries receive similar sARE values, the majority of query values vary, implying that the predictors succeed and fail based on the query.

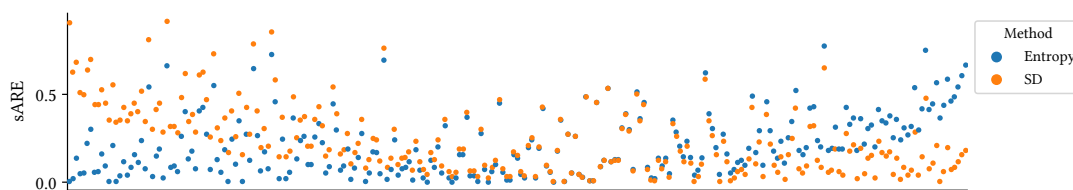


Figure 3: sARE values per query for the NeuralRanker. The queries (x-axis) are sorted by the difference in sARE between *entropy* and *SD*. In the middle are queries that were predicted similarly by both QPPs, and on each side are queries that were predicted well by one predictor and poorly by the other. Each QPP method was applied with the optimal k .

Discussion: Entropy in neural networks. Cross-entropy loss is one of the most common optimization objectives used in neural networks. Neural LMs (e.g., BERT) are trained to minimize cross-entropy loss during pre-training. Many learning-to-rank losses, such as the pointwise binary cross-entropy loss, pairwise RankNet loss [22], and listwise ListNet loss [23], are derived from cross-entropy loss. In short, neural networks are likely to be naturally entropy-aware as they are optimized to *minimize entropy*. During training, the smaller the entropy is, the better the model is believed to be trained. This should be reflected at inference time as well, and we conjecture that the entropy of the predictions correlates to the model prediction performance. The smaller the inference entropy is, the more robust the model is, and thus it is more likely to be more effective.

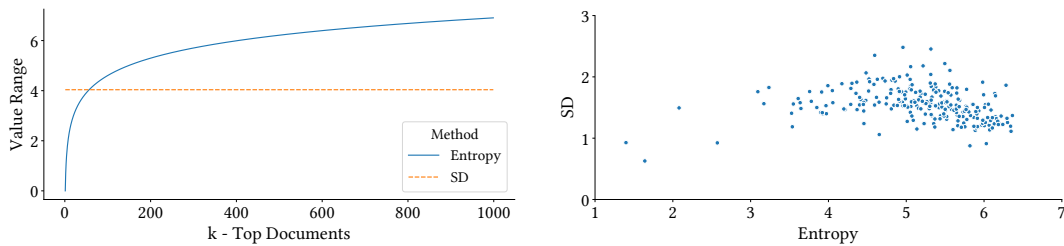


Figure 4: **Left:** shows the value ranges of *entropy* and SD with respect to k . **Right:** shows the actual value spread of *entropy* and SD when $k = 1000$ for 250 queries. The plots were generated for the NeuralRanker RSVs on Robust04.

Discussion: entropy vs SD. Figure 2 shows that the optimal value of the hyperparameter k for SD is relatively small, whereas the optimal k for *entropy* is large in comparison. This could be related to their mathematical properties. Given k scored documents, the *entropy* (of the normalized probabilities) reaches the minimum (zero) when one document has a probability of 1.0 and others have probabilities of 0.0, and reaches the maximum ($\log(k)$) when all the documents have the same probability; SD reaches the minimum of zero when all of the documents have the same score, and has no theoretical upper limit. However, if we know the upper bound b and lower bound a of the scores, *Popoviciu's inequality on variances* shows that the upper limit of SD is $\frac{b-a}{2}$. That is, if we collect a and b for a large number of query-document pairs, we can derive a fixed empirical upper bound of SD. As depicted in Figure 4 (left), the upper SD is fixed and independent of k . Consequently, information is lost by SD as k grows. This can be empirically observed in Figure 4 (right), which shows how a less discriminative SD compares to *entropy* when $k = 1000$. Conversely, the maximum *entropy* increases as k is increased, which suggests that it retains information better than SD for large k . Finally, we note that the notion of *entropy* capturing more information than SD has been observed in other fields [24, 25].

5. Conclusion

This study evaluates the effectiveness of a new QPP method, *entropy*, and compares it to a similar baseline, SD, using a commonly used document retrieval collection Robust04. For Robust04, *entropy* performs similarly to SD when using QL, and outperforms it when using a NeuralRanker retrieval approach. The sensitivity of *entropy* to changes in the hyperparameter k (the number of top-ranked documents that are used in the prediction calculation) is also analyzed, and found to be more robust for *entropy* than for SD. Additional experiments demonstrate that SD and *entropy* typically succeed (or fail) for different queries, suggesting that the methods capture different properties of the distribution of the retrieval scores. Using *entropy* as a feature or in an ensemble model may offer further improvements and would be interesting to explore in future work. Overall, our study provides evidence that *entropy* is a simple and promising method for QPP, and is more robust than currently used methods.

Acknowledgements. We thank the reviewers for their comments. This work was supported by the Australian Research Council's *Discovery Projects* Scheme (grant DP190101113).

References

- [1] S. Cronen-Townsend, Y. Zhou, W. B. Croft, Predicting Query Performance, in: Proc. SIGIR, 2002, pp. 299–306.
- [2] C. Hauff, D. Hiemstra, F. de Jong, A Survey of Pre-Retrieval Query Performance Predictors, in: Proc. CIKM, 2008, pp. 1419–1420.
- [3] H. Roitman, S. Erera, O. Sar-Shalom, B. Weiner, Enhanced Mean Retrieval Score Estimation for Query Performance Prediction, in: Proc. ICTIR, 2017, pp. 35–42.
- [4] O. Zendel, J. S. Culpepper, F. Scholer, Is Query Performance Prediction With Multiple Query Variations Harder Than Topic Performance Prediction?, in: Proc. SIGIR, 2021, pp. 1713–1717.
- [5] O. Zendel, M. P. Ebrahim, J. S. Culpepper, A. Moffat, F. Scholer, Can Users Predict Relative Query Effectiveness?, in: Proc. SIGIR, 2022, pp. 2545–2549.
- [6] Y. Zhou, W. B. Croft, Query Performance Prediction in Web Search Environments, in: Proc. SIGIR, 2007, pp. 543–550.
- [7] A. Shtok, O. Kurland, D. Carmel, F. Raiber, G. Markovits, Predicting Query Performance by Query-Drift Estimation, *ACM Trans. Inf. Sys.* 30 (2012) 1–35.
- [8] S. Datta, D. Ganguly, M. Mitra, D. Greene, A Relative Information Gain-Based Query Performance Prediction Framework With Generated Query Variants, *ACM Trans. Inf. Sys.* 41 (2022).
- [9] G. Faggioli, T. Formal, S. Marchesin, S. Clinchant, N. Ferro, B. Piwowarski, Query Performance Prediction for Neural IR: Are We There Yet?, in: Proc. ECIR, 2023, pp. 232–248.
- [10] Y. Tao, S. Wu, Query Performance Prediction by Considering Score Magnitude and Variance Together, in: Proc. CIKM, 2014, pp. 1891–1894.
- [11] J. Pérez-Iglesias, L. Araujo, Ranking List Dispersion as a Query Performance Predictor, in: Proc. ICTIR, 2009, pp. 371–374.
- [12] J. Pérez-Iglesias, L. Araujo, Standard Deviation as a Query Hardness Estimator, in: Proc. SPIRE, 2010, pp. 207–212.
- [13] R. Cummins, J. Jose, C. O’Riordan, Improved Query Performance Prediction Using Standard Deviation, in: Proc. SIGIR, 2011, pp. 1089–1090.
- [14] H. Roitman, S. Erera, B. Weiner, Robust Standard Deviation Estimation for Query Performance Prediction, in: Proc. ICTIR, 2017, pp. 245–248.
- [15] O. Zendel, A. Shtok, F. Raiber, O. Kurland, J. S. Culpepper, Information Needs, Queries, and Query Performance Prediction, in: Proc. SIGIR, 2019, pp. 395–404.
- [16] J. Lafferty, C. Zhai, Document Language Models, Query Models, and Risk Minimization for Information Retrieval, in: Proc. SIGIR, 2001, pp. 111–119.
- [17] B. Liu, H. Zamani, X. Lu, J. S. Culpepper, Generalizing Discriminative Retrieval Models Using Generative Tasks, in: Proc. WWW, 2021, pp. 3745–3756.
- [18] J. S. Culpepper, B. Liu, RMIT at TREC Deep Learning Track 2020, in: Proc. TREC, 2020, p. 5.
- [19] E. M. Voorhees, The TREC Robust Retrieval Track, *SIGIR Forum* (2005) 11–20.
- [20] G. Faggioli, O. Zendel, J. S. Culpepper, N. Ferro, F. Scholer, An Enhanced Evaluation Framework for Query Performance Prediction, in: Proc. ECIR, 2021, pp. 115–129.
- [21] G. Faggioli, O. Zendel, J. S. Culpepper, N. Ferro, F. Scholer, sMARE: A New Paradigm to

- Evaluate and Understand Query Performance Prediction Methods, *Inf. Retr.* (2022).
- [22] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, G. Hullender, Learning to Rank Using Gradient Descent, in: *Proc. ICML*, 2005, pp. 89–96.
 - [23] Z. Cao, T. Qin, T. Liu, M. Tsai, H. Li, Learning to Rank: From Pairwise Approach to Listwise Approach, in: *Proc. ICML*, 2007, pp. 129–136.
 - [24] G. C. Philippatos, C. J. Wilson, Entropy, Market Risk, and the Selection of Efficient Portfolios, *Applied Economics* 4 (1972) 209–220.
 - [25] S. R. Bentes, R. Menezes, Entropy: A New Measure of Stock Market Volatility?, *Journal of Physics: Conference Series* 394 (2012) 012033.