A Relative Information Gain-based Query Performance Prediction Framework with Generated Query Variants

SUCHANA DATTA, University College Dublin, Ireland
DEBASIS GANGULY, University of Glasgow, United Kingdom
MANDAR MITRA, Indian Statistical Institute, India
DEREK GREENE, University College Dublin, Ireland

Query performance prediction (QPP) methods, which aim to predict the performance of a query, often rely on evidences in the form of different characteristic patterns in the distribution of Retrieval Status Values (RSVs). However, for neural IR models, it is usually observed that the RSVs are often less reliable for QPP because they are bounded within short intervals, different from the situation for statistical models. To address this limitation, we propose a model-agnostic QPP framework that gathers additional evidences by leveraging information from the characteristic patterns of RSV distributions computed over a set of *automatically-generated* query variants, relative to that of the current query. Specifically, the idea behind our proposed method - Weighted Relative Information Gain (WRIG), is that a substantial relative decrease or increase in the standard deviation of the RSVs of the query variants is likely to be a relative indicator of how easy or difficult the original query is. To cater for the absence of human-annotated query variants in real-world scenarios, we further propose an automatic query variant generation method. This can produce variants in a controlled manner by substituting terms from the original query with new ones sampled from a weighted distribution, constructed either via a relevance model or with the help of an embedded representation of query terms. Our experiments on the TREC-Robust, ClueWeb09B and MS MARCO datasets show that WRIG, by the use of this relative changes in QPP estimate, leads to significantly better results than a state-of-the-art baseline method which leverages information from (manually created) query variants by the application of additive smoothing [64]. The results also show that our approach can improve the QPP effectiveness of neural retrieval approaches in particular.

CCS Concepts: • **Information systems → Query intent**; **Information retrieval query processing**.

Additional Key Words and Phrases: Query Performance Prediction, Neural Model Retrieval Scores, Query Variant Generation

ACM Reference Format:
Suchana Datta, Debasis Ganguly, Mandar Mitra, and Derek Greene. 2022. A Relative Information Gain-based Query Performance Prediction Framework with Generated Query Variants . *ACM Transactions on Information Systems* 1, 1 (June 2022), 31 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Query performance prediction (QPP) remains an active area of research in Information Retrieval (IR), primarily because of its usefulness in estimating whether the top-retrieved documents satisfy the

Authors' addresses: Suchana Datta, University College Dublin, Dublin, 4, Ireland, suchana.datta@ucdconnect.ie; Debasis Ganguly, University of Glasgow, Glasgow, United Kingdom, Debasis.Ganguly@glasgow.ac.uk; Mandar Mitra, Indian Statistical Institute, Kolkata, India, mandar@isical.ac.in; Derek Greene, University College Dublin, Dublin, Ireland, derek.greene@ucd.ie.

1046-8188/2022/6-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

underlying information needs of queries without requiring the availability of relevance assessments. This is particularly important because the retrieval effectiveness of IR models can vary substantially for queries with different characteristics [64], spanning from specific to generic [11], or from short to verbose [27].

To introduce the notion of QPP, it represents a class of automated methods that facilitates an IR model to retrospect on its retrieval quality for a given query without the presence of relevance assessments [22]. A QPP method may thus enable an IR system to use this estimate to retrieve more relevant information by applying a number of additional processing steps, either in a user-agnostic or in a user-engaging manner. Instances of user-agnostic processing include selective application of pseudo-relevance feedback [8, 50] involving the automatic augmentation of a user's initial query to retrieve more informative content during a subsequent retrieval step [36, 40, 52, 63]. Methods requiring user engagement include query suggestion [39], or presenting the user with a list of potentially useful query reformulations [2, 24, 37, 44]. QPP methods are intended to allow a selective application of these user-agnostic or user-aware processing steps to further improve the quality of the retrieved information for those queries for which a QPP method estimates a low likelihood of success in finding relevant information [50].

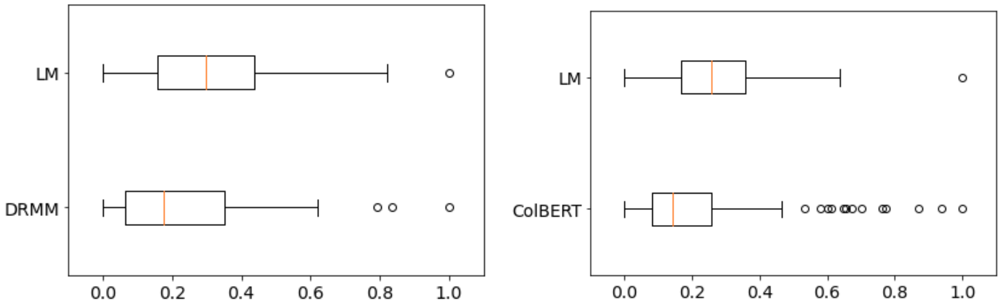
In general, a QPP method estimates the likelihood of relevance of the top-retrieved documents by measuring the distinctiveness of the information need of the current query with respect to the overall topic distribution of the collection. In other words, a QPP method estimates how feasible it is to topically separate the top-retrieved documents from the rest of the collection [28, 53, 56, 62, 70].

Recently, supervised deep neural ranking models have been shown to improve retrieval effectiveness as compared to their unsupervised statistical counterparts [18, 20, 26, 33, 34, 60]. In contrast to human engineered similarity heuristics (e.g., relative term frequency and IDF in BM25), these supervised models rely on a completely data-driven approach of *learning* these similarity functions for ranking documents. These supervised models either typically leverage an early interaction mechanism by computing the similarities between the word vectors of queries and top-retrieved documents [18, 26, 60], or alternately applying a late interaction between the queries and the documents to minimize a triplet-based ranking loss function [34].

However, applying off-the-shelf QPP estimators on neural ranking models is likely to yield limited QPP effectiveness (and this is something that we confirm via our experiments reported later in this paper). This is likely due to the inherently different ways in which the similarity scores or the retrieval status values (RSVs) are computed in the supervised neural models, as compared to their traditional statistical counterparts, e.g., BM25 [45], or Language Model (LM) [43, 65, 66]. Specifically, RSVs in a neural model are computed via the application of a *neural activation function*, such as `tanh` or `relu` [25]. The range of these neural activation functions, and hence the value of a document score (RSV), is thus strictly bounded within a short interval (e.g., $\tanh : \mathbf{x} \mapsto [-1, 1]$ and $\text{relu} : \mathbf{x} \mapsto [0, 1], \mathbf{x} \in \mathbb{R}^d$). This is characteristically different from traditional statistical models, where these bounds are not fixed. In the latter case, they rather depend on the maximum and the minimum values of the term weights in documents and the collection statistics of terms across the collection [29]. Due to the use of the non-linear activation functions, the features that are typically useful for QPP approaches, e.g., the variance [56] or information gain [70] of the document scores, are expected to be less reliable for supervised neural models.

As an illustrative example, Figure 1 compares the distribution of the NQC values (variances of the RSVs) obtained on the TREC-Robust and TREC-DL query sets using two different retrieval models - an unsupervised statistical model and a neural one. More specifically, the statistical model used here is LM with Dirichlet smoothing [65] (henceforth abbreviated as LM-Dir), and the neural

Fig. 1. Comparing the distribution of NQC (a QPP method) scores, normalized in [0, 1], for LM-Dirichlet [65] (a statistical model) and two neural models, DRMM [26] (left) and ColBERT [34] (right) on the TREC-Robust and the TREC-DL topic sets, respectively. It can be seen that the NQC scores for the neural models exhibit a heavier tail.



models used are the DRMM (Deep Relevance Matching Model) [26] and ColBERT (Contextualized Late Interaction over BERT) [34].

Figure 1 shows that not only do the QPP estimates of neural models exhibit a heavier tail as evident from the median line shifted towards the left, but they are also restricted to a much smaller range (compare the spans of the box plots). This behaviour is likely to make it more challenging to effectively estimate the QPP scores, or in other words, effectively distinguish between the queries for which a retrieval model performs well and those for which it does not.

Contributions of this research. We propose an unsupervised post-retrieval QPP estimator, which we refer to as **Weighted Relative Information Gain (WRIG)**. This approach is particularly targeted at neural re-rankers, where the only inputs available to a QPP estimator are the RSVs computed by neural activation functions. Since WRIG relies only on the RSVs obtained from a (neural) model and does not make any specific assumptions about the model architecture itself, it can be applied to the output of any neural, or in fact, any statistical IR model.

To alleviate the limitation in the diversity and range of the RSVs obtained from a neural model, for a given query Q, first we *automatically generate* a set of equivalent queries with a similar information need, which we call $\mathcal{E}_Q$. We then retrieve documents for these query variants and subsequently characterize the retrieval quality of the original query Q with an increase (or decrease) in the aggregated QPP estimate of the variants relative to that of Q itself. To summarise, the novel contributions of this research are:

(1) To the best of our knowledge, this is the first proposal for a generic framework for QPP that leverages information from equivalent expressions of information needs, where there is no requirement on the availability of pre-existing query variants (unlike. e.g., [64]), thus making our proposed method more appealing from a pragmatic point-of-view.
(2) This is the first comprehensive study involving comparisons between statistical and neural models. There do exists neural supervised approaches that estimate QPP for statistical models [19, 62], and also supervised approaches that estimate QPP for neural models [3], but our work is different from both these threads. More precisely, we study the application of an *unsupervised* QPP approach that is particularly appropriate for neural models, although our model is generic enough to be applied even to statistical models.

The remainder of the paper is organized as follows. Section 2 reviews related work on QPP. After establishing the prerequisites in Section 3, we describe our proposed method in Sections 4 and

5. We then present the experimental setup in Section 6, which is followed by a presentation and analysis of the results in Section 7. Finally, in Section 8 we conclude with directions for future work.

2 RELATED WORK

The problem of query performance prediction (QPP) has been widely studied in the literature over a number of years [9, 14, 15, 17, 30, 35, 47, 54, 56, 58, 69, 70]. Generally speaking, QPP is intended to automatically estimate the retrieval effectiveness of a query without relying on relevance judgments [22, 61]. Instead, a QPP method typically relies on two broad sources: *i) pre-retrieval* information, which is available from the collection statistics of an index; and *ii) post-retrieval* information, which becomes available only after a top-set of documents is actually retrieved from an indexed collection in response to a given query.

2.1 Pre-retrieval approaches

A pre-retrieval estimator uses aggregated collection-level statistics (e.g., maximum or average of the inverse document frequencies of the query terms) as a measure of the QPP estimate of an input query. This is based on the assumption that queries with higher QPP estimates are likely to lead to a more topically-coherent set of top-documents [29, 31, 68], and hence are likely candidates for effective retrieval. More recently, a pre-retrieval QPP approach that makes use of the clustering hypothesis of the embedded space of word vectors was proposed in [53]. This method assumes that a query is more specific (hence potentially yielding better retrieval effectiveness) if the cluster membership of the word vectors in the neighborhood of the query terms exhibit a relatively non-uniform distribution.

2.2 Post-retrieval unsupervised approaches

A post-retrieval estimator, on the other hand, makes use of the information from the set of top-retrieved documents to estimate how topically distinct are the top-retrieved documents from the rest of the collection, a large difference indicating potentially better retrieval quality [14]. Various evidences extracted from the top-retrieved documents have been shown to be useful for different post-retrieval QPP estimation methods. This includes those of the KL divergence between the language model of the top-retrieved documents and the collection model in Clarity [14], the aggregated values of the information gains of each top-retrieved document with respect to the collection in WIG (Weighted Information Gain) [70], the skew of the RSVs measured with variance in NQC (Normalized Query Commitment) [56], and ideas based on the clustering hypothesis for a pairwise document similarity matrix [22].

Among ensemble-based approaches, it has been shown that a linear combination of different QPP estimators yield improvements over the individual performance of each [35, 53]. This is somewhat analogous to the use of fusion in retrieval models to yield better retrieval performance [7].

Among the different ways of utilizing RSVs for post-retrieval QPP estimation, assessing the standard deviation of retrieval scores has consistently been employed as an indicator of query performance [17, 42, 56, 57]. It has been observed that the higher the standard deviation, the lower the chances of a query drift [10, 56]. This has led researchers to improve the estimation of standard deviation by applying a bootstrap sampling approach to the top-retrieved list [49]. Another work in this area revisited the estimation of NQC, claiming that NQC computation can be derived as a scaled calibrated-mean estimator [48], which is, in fact, employed as a baseline in this paper.

2.3 Unsupervised approaches involving Evidence Combination

As an alternative to statistical QPP approaches, which leverage information from a single set of top-retrieved documents, there also exists a thread of work that uses a decision theory-based approach for optimally aggregating the evidences from a number of samples drawn from the top-retrieved document sets for the purpose of achieving a more robust QPP estimate [35, 46, 51, 54].

Another line of research has shown that using information from reference queries (i.e., those that possess a similar information need to that of the original query) can improve QPP estimates. These reference queries or query variants are either manually created (i.e. extracted from search sessions for computational purposes [5]) or are automatically created by a term association based approach such as the relevance model (RLM) [47, 55].

Among these reference list based QPP approaches, we utilize a recent method - RLS [47] as one of the baselines in this paper. Specifically, the RLS method estimates the performance of a query by making use of information from lists of documents retrieved with a number of augmented queries. The basic difference of RLS with our proposed method is that we generate query variants by substituting, in general, a multiple number of terms (more details in Section 5), whereas the query augmentation process in RLS [47] involves adding only a single term.

While these reference-list based approaches generally aim to predict the effectiveness of the initial result list by taking into account the additional reference list of queries, the study in [51] attempted to predict the quality of a second-stage retrieval step obtained via relevance feedback. Since the focus of this paper is to investigate QPP for neural models, which usually involve a re-ranking step similar to [51], we employed this pseudo-feedback based QPP method as one of our baselines as well.

A major difference of our proposed method from [51] is that our model makes use of *only the RSVs* obtained by the neural re-rankers, whereas PFR-QPP - the method of [51], leverages information from both the feedback and the initial result lists (more details in Section 3.1).

Zendel et al. [64] reported improvements in QPP effectiveness by using a set of manually generated query variants. In particular, their method involved applying a linear smoothing technique to combine estimated scores obtained from other variants into an estimated score for the original query. Our method differs from [64] in two important ways. First, in contrast to the additive smoothing-based approach, our method employs relative differences, and more importantly, second, as a part of our proposed framework, we *automatically generate* the alternative expressions of the information need of a query, which means that the use-case of our method is not restricted by the availability of manually-formulated query variants.

2.4 Supervised approaches

Among supervised approaches, the authors of [62] proposed a weakly supervised neural approach to learn the relative importance of different estimators to find an optimal combination. In contrast to weak supervision of [62], end-to-end supervised QPP approaches were proposed in [3] and [19]. These approaches seek to learn a functional association between the input data (query-document interaction) and the ground-truth values of retrieval effectiveness measures on a training set of queries.

The main difference between our work in this paper and the previously proposed supervised approaches is that we propose an *unsupervised QPP method for supervised ranking models*. In fact, due to this reason we do not compare our proposed unsupervised model with the supervised QPP approaches existing in the literature [3, 19, 62].

3 PREREQUISITES

Before describing the details of our proposed QPP method in Section 4, we first discuss the necessary prerequisites, specifically relating to existing QPP methods and neural re-rankers.

3.1 An overview of post-retrieval QPP estimators

In this section, we introduce a generic framework for post-retrieval QPP that makes use of query variants. Standard QPP estimators are first established as special cases in the framework – ones that do not make use of the variants. In general, given a query, Q, a post-retrieval QPP method estimates the probability of successfully retrieving useful information in response to Q, $P(S|Q)$, as a function Φ of the query itself and its top-k retrieved document set M_k, i.e.,

$$P(S|Q) \approx \Phi(Q, M_k(Q)),\ \ M_k = \{D_i\}_{i=1}^{k}. \tag{1}$$

Existing post-retrieval QPP methods use different forms of the function $\Phi(Q, M_k(Q))$. We now describe a number of such forms.

Normalized Query Commitment (NQC) [56]. This is a commonly used post-retrieval QPP method that predicts the retrieval effectiveness of a query using the standard deviation of the document scores. This follows the hypothesis that a query with a well-defined information need is likely to lead to a more non-uniform (heavy-tailed) distribution of the RSVs. To compute the variance of the RSVs in NQC, the function Φ of Equation 1 takes the form

$$\Phi_{\text{NQC}}(Q, M_k(Q)) \overset{\text{def}}{=} \frac{\sqrt{\frac{1}{k}\sum_{i=1}^{k}(P(D_i|Q) - \bar{P}(D|Q))^2}}{P(Q|C)}, \tag{2}$$

where $P(D_i|Q)$ denotes the similarity score of the document D_i to Q, $\bar{P}(D|Q)$ denotes the mean of the RSVs, and $P(Q|C)$ denotes the similarity of Q to the collection, which is computed by aggregating collection statistics over the query terms.

Scaled Calibrated NQC (SCNQC) [48]. This model is a generalization of NQC which involves a number of parameters, both in terms of calibration and scaling. The optimal values of these parameters are found by a coordinate ascent or a grid-based exploration. This measure is formally written as

$$\Phi_{\text{SCNQC}}(Q, M_k(Q)) \overset{\text{def}}{=} \frac{1}{k}\sum_{i=1}^{k}\left[P(D_i|Q)\left(\frac{1}{P(Q|C)}\right)^{\alpha}\left(\frac{P(D_i|Q) - \bar{P}(D|Q)}{\sqrt{P(D_i|Q)}}\right)^{\beta}\right]^{\gamma}, \tag{3}$$

where the expressions $P(D_i|Q)$, $\bar{P}(D|Q)$, and $P(Q|C)$ carry the same meaning as in Equation 2. Additionally, α is an idf-weighting factor, β is a weighting factor associated with the deviations in scores and γ is a calibration parameter.

Weighted Information Gain (WIG) [70]. WIG uses the aggregated value of the information gain of each top-retrieved document with respect to the collection. The more topically distinct a document is from the collection, the higher its gain will be. This means that the underlying hypothesis of WIG is mostly similar to that of NQC. The average of these information gains characterizes how topically distinct the overall set of top-documents is. Formally,

$$\Phi_{\text{WIG}}(Q, M_k(Q)) \overset{\text{def}}{=} \frac{1}{|M_k(Q)|}\sum_{D \in M_k(Q)} \frac{1}{\sqrt{|Q|}}\sum_{q \in Q} \log P(D|Q) - \log P(q|C), \tag{4}$$

where $P(D|Q)$ denotes the score of a document D with respect to the query Q, and $P(q|C)$ denotes the collection statistics of a query term $q \in Q$. Zhou and Croft [70] proposed the use of $1/\sqrt{|Q|}$ as a normalization constant so that the WIG scores across queries of different lengths become comparable.

Clarity [14]. This method estimates a relevance model (RLM) [36] distribution of term weights from a set of top-ranked documents and then computes its KL divergence with the collection model. The hypothesis is that higher the KL divergence score is, the higher is the QPP estimate. For estimating the clarity score of a query Q, the generic function Φ of Equation 1 takes up the following form.

$$\Phi_{\text{Clarity}}(Q, M_k(Q)) \overset{\text{def}}{=} \sum_{w \in V_{M_k(Q)}} P(w|\theta_{M_k(Q)}) \log \frac{P(w|\theta_{M_k(Q)})}{P(w|\theta_C)}, \tag{5}$$

where C denotes the collection, $M_k(Q)$ denotes the set of top-k retrieved documents for a query Q, $V_{M_k(Q)}$ is the vocabulary of $M_k(Q)$, and $\theta_{M_k(Q)}$ and θ_C are, respectively, the relevance model estimated from $M_k(Q)$, and the language model of the collection.

UEF [54]. Different from the estimators discussed so far in this section, the UEF method involves estimating a confidence score for a set of top documents itself, assuming that the value of the estimator itself is potentially more reliable for certain sets of top-retrieved documents than others. As a first step, the UEF method estimates how robust a set of top-retrieved documents is by checking the relative stability in the rank order before and after relevance feedback (e.g., by RLM). The higher the perturbation of a ranked list is following the feedback operation, the greater is the likelihood that the retrieval effectiveness of the initial list was poor, which in turn suggests that a smaller confidence should be associated with the QPP estimate of such a query. Formally,

$$\Phi_{\text{UEF}}(Q, M_k(Q), \phi) \overset{\text{def}}{=} \sigma(M_k(Q), M_k(\theta_Q))\phi(Q, M_k(Q)), \tag{6}$$

where $\phi(Q, M_k(Q))$ is, as per the terminology of [54], a 'base QPP estimator' (e.g. WIG or NQC), $M_k(\theta_Q)$ denotes the re-ranked set of documents post-RLM feedback, the RLM being estimated on the initially retrieved set of top-k documents $M_k(Q)$, and σ is a rank correlation coefficient (e.g. Spearman's ρ or Kendall's τ) of two ordered sets.

PFR-QPP [51]. The PFR-QPP method estimates the QPP effectiveness on a second-stage retrieved list of documents (usually obtained via relevance feedback). The method involves estimating the QPP score of the second-stage retrieval as a combination of two different scores: a) an independent estimate of the second list, and b) its estimation conditioned on the initial retrieval. Formally,

$$\Phi_{\text{PFR-QPP}}(Q, M_k(Q), \theta) \overset{\text{def}}{=} \left[P(M_k(\theta_Q), \theta)\right]^{\eta} \left[P(M_k(\theta_Q), M_k(Q), \theta)\right]^{(1-\eta)}, \tag{7}$$

where, $M_k(Q)$ is the initial retrieval list of top-k pseudo-relevant documents for the query Q, θ and $M_k(\theta_Q)$ denote the relevance model estimated from $M_k(Q)$ and re-retrieved list obtained by θ, respectively. Additionally, η acts as a parameter controlling the relative importance of the two different estimators (see [51] for additional details on how these QPP components are computed).

3.2 QPP using reference queries

Any statistical estimation method can, in principle, be improved with the availability of a large number of observation points. In the context of QPP, since a post-retrieval estimator relies on the computation of statistical measures (e.g., the variance of the RSVs in NQC), the estimate for a query Q can be improved by leveraging information from other queries similar to Q. The post-retrieval estimator of Equation 1 can thus be further generalized as

$$P(S|Q, \mathcal{E}_Q) \approx \Phi^{+}(Q, \mathcal{E}_Q, M_k(Q), \cup_{Q' \in \mathcal{E}_Q} M_k(Q')), \tag{8}$$

where the function Φ^+ is a generalization of the function Φ of Equation 1 with additional parameters, namely $\mathcal{E}_Q$ and $\cup_{Q' \in \mathcal{E}_Q} M_k(Q')$. In particular, $\mathcal{E}_Q$ denotes a set of expressions of information need equivalent to that of Q and $\cup_{Q' \in \mathcal{E}_Q} M_k(Q')$ represents the top-documents retrieved with each query Q' in this set $\mathcal{E}_Q$. The effect of these additional parameters is that not only does Φ^+ depend on the top-retrieved list for Q, but it is also characterized by $\mathcal{E}_Q$ and the top-list retrieved for each query in this set.

As a concrete realization of the generic function Φ^+ of Equation 8, the authors of [64] proposed to use linear smoothing. More precisely, the QPP estimate for a query is combined with the QPP estimate from other similar queries. Formally,

$$P(S|Q, \mathcal{E}_Q) = (1-\lambda)\Phi(Q, M_k(Q)) + \frac{\lambda}{|\mathcal{E}_Q|} \sum_{Q' \in \mathcal{E}_Q} \Phi(Q', M_k(Q'))\sigma(Q, Q'), \quad (9)$$

where λ is a smoothing parameter, Φ represents a generic QPP estimator (NQC being specifically used in [64]), and $\mathcal{E}_Q$ denotes the set of *equivalent* queries, also known as *query variants* or reference queries [5, 6, 12, 64]. The factor $\sigma(Q, Q')$ in Equation 9 denotes a relative contribution from each variant, allowing the provision for the information from some variants to be more reliable than others. The study [64] investigated different ways of considering the similarity between a query Q and its variant Q' and reported that the rank-biased overlap (RBO) [59] is the most effective way of accounting for this relative weight, among other alternatives, such as Jaccard similarity between query terms or the similarity between the sets $M_k(Q)$ and $M_k(Q')$.

3.3 Neural models

In this paper, we provide a brief introduction to how neural models work and argue why off-the-shelf QPP approaches may fail to work well for these models. In particular, for our experiments we use two query-document interaction-based neural re-rankers with largely different characteristics, namely (i) Deep Relevance Matching Model (DRMM), the *early interaction-based* model where the combined information from the embeddings of a query and a document is passed on to a feed-forward network [26], and (ii) ColBERT, the *late interaction-based* model, where the interaction takes place at a much later stage between the encoded representation of the constituent terms of a document and a query [34].

Supervised neural models are generally trained in a pairwise manner to minimize a triplet loss of the form

$$\mathcal{L}(Q, D_r, D_n) = \sigma(Q \odot D_r; \theta) - \sigma(Q \odot D_n; \theta), \quad (10)$$

where D_r represents a relevant document and D_n denotes a non-relevant one.

The objective of the loss function is to learn the optimal representation of the interaction vector (parameterized by the set of θ matrices) so as to maximize, on the one hand, the query's similarity with a relevant document, and minimize its similarity with a non-relevant document on the other. We now explain each component of Equation 10 in the subsequent part of this section.

The function $\odot : (Q, D) \mapsto \mathbb{R}^p$ in Equation 10 represents an interaction operation between a query and a document, that outputs a vector of a fixed dimension. For instance, in DRMM, this maps a query term and a document in a histogram indicating the number of times the cosine similarity between a given query term and a constituent term of a document D falls within a quantized interval of $[-1, 1]$.

The other function $\sigma(\mathbf{x}; \theta) \mapsto \mathbb{R}$ is a linear[1] function parameterized by the learnable set of parameters θ. In general, the matrix θ represents the parameters of a feed-forward network with

[1] The activation of each neuron, however, is a non-linear function, e.g. the sigmoid.

$l \geq 1$ layers, in which case, $\theta = \{\theta_{(1)}, \ldots, \theta_{(l)}\}$, such that the outputs of the intermediate layers are given by $\sigma_{(i)} = \theta^T_{(i)} \cdot \sigma_{(i-1)}$ with $\sigma_{(1)} = \theta^T_{(1)} \cdot (Q \odot D)$ denoting the output of the first layer.

It is worth mentioning that the output from the final layer of a network (and also those of the intermediate layers) are usually bounded within the range of the activation function used. For example, $\sigma(\mathbf{x}; \theta) \mapsto [-1, 1]$ if the activation employed in the parameterized linear function of Equation 10 is 'tanh' (likewise, with 'sigmoid' the range becomes $[0, 1]$).

In ColBERT the interaction operator takes a different form in the sense that the function $\odot$ in Equation 10 corresponds to the sum of maximum cosine similarities between the encoded representations of the constituent terms between a query and a document. In particular, ColBERT computes the relevance score as a sum over the maximum cosine similarity values obtained from the query-document BERT embeddings, i.e., the score takes the form of $\sigma(\mathbf{x}; \theta) : \sum_{i \in Q} \max_{j \in D} \mathbf{v}_{Q_i} \mathbf{v}^{\mathrm{T}}_{D_j} \mapsto [0, \infty]$, where $\mathbf{v}_Q$ and $\mathbf{v}_D$ are the BERT [21] embeddings of a query Q and a document D, respectively.

From Equation 10, it can be realized that the scores obtained with a neural model are characteristically different from those obtained with statistical models. While the RSVs in DRMM is essentially restricted within $[0, 1]$, for ColBERT they usually occupy a wider range (as the ColBERT score is an aggregation over the pairwise cosine similarities between query-document terms). However, these ColBERT scores when compared with the RSVs of a statistical model, are still restricted within a much shorter range. This behaviour of the neural models may eventually limit the effectiveness of off-the-shelf QPP approaches. With this background, in the next section we delve into the details of our proposed approach.

4 WEIGHTED RELATIVE INFORMATION GAIN-BASED MODEL - WRIG

4.1 Motivation

The method previously proposed in [64] for estimating QPP with query variants uses the RSVs obtained from statistical models, such as BM25 or LM. The method itself (Equation 9) does not make any specific assumptions on the range of the RSVs. However, unlike the RSVs of statistical models, the similarity scores from a neural reranking model are essentially parameterized; for instance, compare the function $\Phi(Q, D)$ of Equation 1 with $\sigma(Q, D; \theta)$ of Equation 10. Moreover, the final output value of a network (and also those of the intermediate layers) are in fact necessarily bounded within the range of the activation function used. For example, $\sigma(\mathbf{x}; \theta) \mapsto [-1, 1]$ if the activation employed in the parameterized linear function of Equation 10 is tanh.

Therefore, due to the strictly bounded nature of the RSVs, an RSV-based post-retrieval QPP estimator, such as NQC (Equation 2), may not be effective in predicting retrieval quality for a query. In fact, our experiments with standard QPP approaches confirm this hypothesis. In Section 7, we show that there is a substantial difference between the effectiveness of standard QPP approaches when applied on statistical vs. neural ranking models.

Our initial experiments showed a similar trend for a state-of-the-art QPP approach [64] that relies on augmenting information from (manually created pre-existing) query variants. Even though this method had reported to improve QPP effectiveness [64], our experiments show that:

- The method proposed in [64] is substantially less effective for neural models than for statistical models (we discuss this later in Section 7).
- Even worse, a straightforward application of the QPP method of [64] leads to a decrease in the QPP effectiveness for neural models with respect to standard baselines. We discuss more about this observation in Section 7.

Motivated by these observations, we now propose a method that seeks to use additional data from query variants in a manner that is different from that of the additive smoothing based technique. As in [64], we first describe our QPP method assuming that the variants of a query are available

Table 1. A contingency table demonstrating the four possible cases of QPP estimation with the method of relative differences. The relative ratio of QPP difference, $\Delta\Phi(Q, \mathcal{E}_Q)$, is computed as $(\Phi(Q, M_k(Q)) - \bar{\Phi}(\mathcal{E}_Q))/\Phi(Q, M_k(Q))$ (see Equation 11). The warmth of a color indicates the QPP estimate of Q, whereas the intensity of a color denotes the confidence in the QPP estimation.

		Magnitude of $\Delta\Phi(Q, \mathcal{E}_Q)$	
		High	Low
Sign of $\Delta\Phi(Q, \mathcal{E}_Q)$	> 0	QPP estimate↑	QPP estimate↑
	≤ 0	QPP estimate↓	QPP estimate↓

to the QPP estimator. Later, in Section 5, we describe two methods to automatically generate an effective set of query variants. This is particularly important in cases where either query variants are unavailable, or manually generating variants is prohibitively time consuming.

4.2 Relative Differences in QPP estimate

Instead of using additive smoothing from the likelihood of QPP estimate of query variants of [64] (Equation 9), we propose a different realization of the generic function Φ^{+} of Equation 8. In particular, we first compute the estimated likelihoods of QPP estimate of these variants, after which we compute the relative difference in the expected likelihood (average value) of the QPP estimate of the variants with respect to that of the given query itself. Formally speaking,

$$\begin{aligned} P(S|Q, \mathcal{E}_Q) &= \Delta\Phi(Q, \mathcal{E}_Q) = \frac{\Phi(Q, M_k(Q)) - \bar{\Phi}(\mathcal{E}_Q)}{\Phi(Q, M_k(Q))}, \\ \bar{\Phi}(\mathcal{E}_Q) &= \frac{1}{\sum\limits_{Q' \in \mathcal{E}_Q} \sigma(Q, Q')} \sum_{Q' \in \mathcal{E}_Q} \Phi(Q', M_k(Q'))\sigma(Q, Q'). \end{aligned} \tag{11}$$

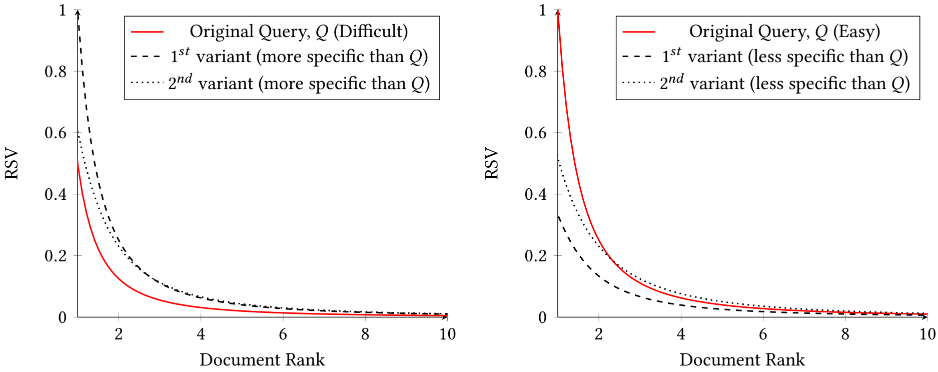
Equation 11 can be interpreted as follows. A large value of the predictor for the original query, $\Phi(Q, M_k(Q))$ in conjunction with a small average value of the predictor for the variants, $\bar{\Phi}(\mathcal{E}_Q)$, means that their relative ratio of difference, $\Delta\Phi(Q, \mathcal{E}_Q)$, is likely to be close to 1. This indicates that the variants are, on average, less specific than the original query. This, in turn, *increases the confidence* of the prediction of the original query to be a specific one.

Likewise, a small value of $\Phi(Q, M_k(Q))$ coupled with a large value of $\bar{\Phi}(\mathcal{E}_Q)$ indicates that the relative ratio of difference $\Delta\Phi(Q, \mathcal{E}_Q)$, should considerably be less than zero. This, in turn, indicates that the variants, on an average, are substantially more specific than the original query, thereby increasing the confidence in predicting Q to be less specific. For the other two cases, i.e. when $|\Delta\Phi(Q, \mathcal{E}_Q)|$ is close to 0, the confidence in prediction is smaller. Table 1 shows a contingency table depicting the four different situations.

We refer to our method of using differences in the QPP estimate of the query variants relative to the original query as Weighted Relative Information Gain (WRIG). The nomenclature reflects the fact that, similar to WIG [70], WRIG uses the concept of *weighted information* as evident from the $\sigma(Q, Q')$ factor of Equation 11. However, the *weights* themselves rather than being interpreted as the contribution of each top-retrieved document in the QPP estimate of the original query Q, are, in fact, reflective of the *relative importance* of each query variant.

Specifically, as per the findings of [64], we make use of the rank-biased overlap (RBO) based similarity [59] between a query Q and its variant Q'. In fact, our experiments demonstrated that the RBO similarity measure outperformed the Jaccard similarity between the query terms and

Fig. 2. A schematic representation of the idea of using the RSV distribution of query variants, $Q' \in \mathcal{E}_Q$ (shown with dotted lines), to estimate the QPP of the current query (Q). **Left**: The non-uniformity (*skew*) of the variants is higher than that of the current query (Q), which means that the QPP estimator predicts a low value of $P(S|Q)$. **Right**: The non-uniformity of Q is higher than those of its variants, in which case our predictor outputs a high $P(S|Q)$.



the cosine similarity between the sets $M_k(Q)$ and $M_k(Q')$ thus corroborating the findings of [64]. Therefore, we only report results with the RBO-based instantiation of $\sigma(Q, Q')$ of Equation 11.

4.3 An Illustrative Example with NQC

While the generic description of WRIG in Section 4.2 involved computing the relative differences with respect to any predictor function $\Phi(Q, M_k(Q))$, we now demonstrate the working principle of WRIG with NQC (i.e., variances of the RSVs) as a particular choice of the estimator function (Equation 2). For instance, substituting the generic estimator function, ϕ of Equation 11 with the NQC estimator yields

$$\begin{aligned} P(S|Q, \mathcal{E}_Q) &= 1 - \frac{1}{\nu(Q,k) \sum_{Q' \in \mathcal{E}_Q} \rho(Q, Q')} \sum_{Q' \in \mathcal{E}_Q} \nu(Q', k)\rho(Q, Q'), \\ \nu(Q, k) &= \mathrm{Var}(\sigma({D^Q}_1), \ldots, \sigma({D^Q}_k)), \end{aligned} \tag{12}$$

where Var denotes the variance function, D_i^Q denotes the i^{th} document retrieved with query Q, $\sigma({D^Q}_i)$ denotes the RSV of the i^{th} document D_i retrieved in response to the query Q, and $\rho(Q, Q')$ measures the RBO-based similarity between the ranked lists retrieved with the variant Q' and the original query Q. Although we used RBO in our experiments as prescribed in [64], it is possible to use any other function to define the similarity measure between the top-retrieved lists of Q' and Q, e.g., Jaccard etc.

Sample distributions of similarity scores for a query with respect to its variants are shown in Figure 2 to illustrate the working principle of WRIG with the variance based NQC estimator. The RSVs provided are in the range of $[0, 1]$, which is the case if either sigmoid or ReLU is used as an activation function for a neural model.

The plot on the left of Figure 2 shows that the RSV distributions of the variants are more skewed (higher variance), in which case the average of the variances aggregated over the set of reference queries is also higher. This means that the sign of the relative change of variance (Equation 11) is negative and the magnitude is high (corresponding to the bottom-right case in the contingency of

Table 1). The NQC-based predictor thus in this case predicts a low QPP estimate for Q. Conversely, the plot on the right shows the situation where the RSVs of the current query are more skewed than those of its variants, thus corresponding to the top-left case in the contingency of Table 1.

4.4 Comparisons with Additive Smoothing

Both the additive smoothing methodology [64] and our proposed relative difference-based approach (Equation 11) use estimates from a set of query variants in addition to the estimated QPP value for the current query. Compared to the additive smoothing approach, the advantage of our method is that it does not involve an additional smoothing parameter, λ, to control the relative importance of the estimated QPP of the original query with respect to its variants.

Another advantage of WRI over additive smoothing is that WRIG allows a more intuitive interpretation of the estimated QPP value of the current query by using the values of the variants as reference points (see Figure 2). For instance, it is not obvious if a query Q with an absolute QPP estimate of $P(S|Q) = 0.6$ qualifies as being a difficult or an easy one. With our proposed method, however, it is possible to interpret this QPP value from the relative perspective of these variants.

Example 4.1. Consider the query ‘Parkinson’s disease’, say with $P(S|Q) = 0.6$, which may seem to be one that is reasonably specific, pointing to a precise information need. However, with respect to one of its variants ‘Parkinson’s disease treatment’, the QPP estimate of which is expected to be higher, say $P(S|Q) = 0.75$, it is possible to conclude that the original query itself was not particularly an easy one to yield sufficiently high retrieval performance. Our method, with reference to this example, would make use of the $(0.75 - 0.6)/0.6 = 25\%$ observed increase in the relative QPP estimate of a variant to eventually help interpret that the original query itself is likely not to be an easy one for an IR system.

4.5 Regression-based QPP Estimation

As a generalized function to compute the non-uniformity of a set of RSVs, we propose to use a linear regression based solution. Assuming that the similarity scores are a function of the document ranks, we estimate the parameters of a line that best fits a given observation – in our case the given set of k pairs of document ranks and scores, i.e. $M_k(Q) = \{(i, P(D_i|Q))\}_{i=1}^{k}$. Formally speaking, we fit a line, parameterized by $\theta \in \mathbb{R}^2$, of the form $\hat{\sigma}(i;\theta) = \theta_1 i + \theta_0$ (i.e. a line with slope of θ_1 and intercept of θ_0) to the observed data, $M_k(Q)$.

It is a well-known result that the closed form solution of the slope of the best fitting line in a two dimensional x-y plane is given by $Cov(X,Y)/Var(X)$ (X and Y denoting the sets of values for the abscissa and the ordinate, respectively). In the context of our problem, we need to compute only the slope of this regressor line, which is computed as

$$\theta_1 = \frac{\sum_{i=1}^{k}(i - \bar{k})(P(D_i|Q) - P(\bar{D}|Q)}{\sum_{i=1}^{k}(i - \bar{k})^2}, \tag{13}$$

where $\bar{k} = (k+1)/2$ denotes the average of the document ranks and $P(\bar{D}|Q) = 1/k \sum_{i=1}^{k} P(D_i|Q)$ denotes the average of the RSVs.

Since the slope of a parametric line indicates the general trend of how rapidly the RSVs decrease over ranks, it is easy to see that the higher the magnitude of the slope, the higher the non-uniformity of the RSVs (i.e., the QPP estimate of a query). As an instance of the linear regression-based predictor function Φ, we therefore use the absolute value of the slope estimated from the fitted document scores. More formally,

$$\Phi_{\mathsf{LR}}(Q, M_k(Q)) \overset{\text{def}}{=} |\theta_1|, \tag{14}$$

where θ_1 is given by Equation 13, and LR denotes linear regression.

Previous studies, such as [4, 16], has applied specific models of statistical distributions, such as the Gamma distribution, Power law distribution or Gaussian Mixture models to fit a given RSV distribution with approaches such as the method of moments (MME) or expectation maximization (EM). Different to these approaches, our method of linear regression to fit the RSV score distribution does not require making any specific assumptions about the inherent nature of the document scores distribution. This makes our estimator a generic one without any specific assumptions about the nature of the retrieval scores produced by a neural model. Off-the-shelf applications of distributions that are known to work well for statistical IR models, such as the Poisson or the Gamma distributions, may not work well for neural models.

5 AUTOMATICALLY GENERATING QUERY VARIANTS

Recall from Section 4 that in contrast to existing QPP approaches, such as NQC [56] or WIG [70], our method relies on the existence of a set of variants or reference queries, similar to the requirement of [64]. However, in practice such reference queries are usually unavailable. Therefore, we explore two different methods for automatically constructing variants from a user's query. Before describing these methods, we first discuss the desirable characteristics of automatically-generated variants.

5.1 Characteristics of the generated variants

Since our goal is to estimate the retrieval quality of a query relative to its variants, the QPP estimate of the variants should not be substantially different from that of the original one. Previous research on query sessions has shown that even one additional term can make a query significantly more specific. In contrast, removing one term can make a query substantially more general, leading to loss of specificity. Returning to Example 4.1, adding the term *treatment* to the query *Parkinson's disease* makes it substantially more specific.

To ensure that the QPP estimate of the variants in WRIG are comparable to that of the original query, while generating the query variants we only allow substituting a randomly chosen term of the original query with another term. The probability of this substitution is given by a distribution of neighboring (semantically related) words to the constituent terms of Q, denoted by $\mathcal{N}(Q)$. More formally,

$$Q' \leftarrow (Q - \{t\}) \cup \{w\} : t \sim Q,\ w \sim \mathcal{N}(Q), \tag{15}$$

where the probability of selecting a term $w \in \mathcal{N}(Q)$ is given by the maximum likelihood estimate over the weights of the terms. We then repeat the sampling step of Equation 15, m number of times, where $1 \leq m \leq |Q| - 1$. This ensures that we substitute m terms from the original query with those sampled from $\mathcal{N}(Q)$, thus ending up retaining $|Q| - m$ terms from the original query with m new related terms being added.

As a word of note, we mention that in our experiments we varied m within the range of 1 to $|Q|-1$, and observed that the effect of m on the final QPP effectiveness measures were non-significant. Hence, we report the results only with the best setting of m, which, as per our observation, was $|Q| - 1$. In other words, the best results were obtained when we retained only a single term from the original query Q.

We now describe two ways to define the set of weighted term distributions from which terms to be substituted are sampled.

5.2 Relevance model-based term substitution

In this case, the set $\mathcal{N}(Q)$ from which related query terms are chosen for substituting an original query term, refers to a distribution of term weights estimated from a standard feedback model, namely the relevance model (RLM) [32, 36]. The weight for a term w in RLM, $P(w|Q, M_k(Q))$, is

Table 2. Examples of automatically-generated query variants for 3 different topics from the TREC-Robust, ClueWeb09B, and TREC-DL datasets respectively. Variants are obtained by substituting terms in the original query with those sampled (biased) from a weighted term distribution, constructed either with relevance feedback (RLM) or with embedded word vectors (W2V).

Dataset	Original Query	Generated Variants	
		RLM	W2V
TREC-Robust	Ireland peace talks qid: 404	Ireland economy paramilitary peace exercise agreement peace talks operation	peace mideastern footdrag talks agreement negotiate talks resume insist
ClueWeb09B	signs of a heartattack qid: 175	heartattack bezoar heartattack motorsport heartattack dormant	sign prognosis heartattack features heartattack seizure
TREC-DL	how long is life cycle of flea qid: 264014	life larva detailed stage flea quickly annihilates control cycle leads female cocoons	flea cycle pupae larva cycle application pupae methoprene long fleas dormant annihilates

estimated by computing the likelihood of the *local* co-occurrences of w with the query terms from the set of top-k retrieved documents, $M_k(Q)$.

Our methodology of query variant generation is a simplification of the method proposed in [12], where the number of terms in the generated query was itself a random integer. In contrast, for our case, the number of terms in each generated variant is identical to the number of terms in the original query. In our experiments, we varied the number of top-selected documents k' for feedback ($\ni k' < k$) in the range of 5 to 20. We observed the best results for $k' = 10$.

5.3 Word embedding-based term substitution

For this query variant generation method, instead of leveraging the relevance feedback based local (top-retrieved) term statistics, we instead define $\mathcal{N}(Q)$ as the union over the set of t nearest neighbors of each query term (in an embedded space of word vectors). Specifically, we used skipgram [38] vectors trained on the part of Google News dataset (about 100 billion words). The model contains 300-dimensional vectors for nearly 3 million words and phrases. Formally,

$$\mathcal{N}(Q) = \cup_{q \in Q}\{w : \mathbf{w} \in \mathcal{N}_t(\mathbf{q})\}, \tag{16}$$

where $\mathcal{N}_t(\mathbf{q})$ denotes the set of t-nearest word vectors relative to the vector for each constituent query term q. The distance function used to define the neighborhood is the cosine distance [38]. In our experiments, we set the values of t in the range from 5 to 20 and found that the best QPP results were obtained with variants generated with 5 nearest neighbors.

Table 2 lists a number of variants generated for 3 different example queries selected from the TREC-Robust, ClueWeb09B and TREC-DL topic sets respectively. We observe that seemingly generic variants, such as ‘peace exercise agreement’ and ‘peace talks operation’, could potentially be useful in WRIG to infer that the original query ‘Ireland peace talks’ is most likely to be a specific one.

6 EXPERIMENT SETUP

In this section, we first describe the specific research questions related to the QPP of neural re-ranking models, following which we describe the datasets and the methods investigated.

6.1 Research questions

As discussed in Sections 1 and 4.1, existing QPP approaches are not expected to work effectively for neural rankers, because the relatively small differences in the scores may pose a difficulty in estimating retrieval effectiveness. Therefore, we formulate the first two research questions:

RQ-1: How well do existing QPP estimators work on neural models? Can a simple approach of applying an *inverse neural activation function* improve QPP effectiveness for neural models?

RQ-2: How effective is our proposed method WRIG of relative difference-based QPP for neural re-ranking models, in comparison to standard post-retrieval QPP approaches?

While existing QPP approaches, such as NQC, WIG etc., are capable of predicting the QPP estimate of a query without the presence of any reference set of other similar queries, the WRIG estimator (Equation 11), essentially relies on the availability of a set of query variants to be able to compute the relative differences. This means that the first step to investigate **RQ-2** is to automatically generate a set of reference queries. In our experiments, we explore two ways of automatically generating reference queries and also compare the QPP effectiveness obtained with manually formulated reference queries (similar to [64], we used the UQV dataset). Our third research question is thus:

RQ-3: Among the local and global approaches for query variants generation (Section 5.2 and 5.3), which one is the most effective for WRIG? What is the relative performance of WRIG with these automatically generated variants as compared to manually formulated ones?

In our fourth research question, the aim is to find the most effective way for WRIG to measure the non-uniformity in the RSVs of the top-retrieved documents:

RQ-4: Among the alternatives of using the variance or linear regression (Equation 14), which one is the most effective for WRIG?

6.2 Settings

6.2.1 Neural model and activation functions. We conduct experiments with two neural rankers of considerably different characteristics (see Section 3.3). The first neural model that we employ is the deep relevance matching model (DRMM). We choose DRMM because being an interaction-driven model, it involves a much smaller number of parameters (usually of the order of 50-100K). This is a likely reason why the model is reported to generalize well for standard ad-hoc test collections with minimal amount of training data [26]. The size of the input for a model like DRMM is relatively small because it uses a quantized interaction operation (histograms of counts of word vector similarities between query and document terms computed over discrete intervals). In contrast, other early interaction-based models, such as KNRM [60], operate on a full matrix of pairwise word vector similarities, and thus the number of parameters in such models is in the order of millions [60]. To explore the effect of our QPP method for different ranges of RSVs, we use two different activation functions, tanh and sigmoid, with corresponding models denoted as $\text{DRMM}_{\text{tanh}}$ and $\text{DRMM}_{\text{sigmoid}}$, respectively.

As our second neural model, we consider the BERT-based late interaction architecture, ColBERT [34]. This model independently encodes the document and the query using BERT [67] and then captures their fine-grained similarities by employing interactions between them (see Section 3.3). We choose ColBERT for our investigation on the effectiveness of QPP because it is one of the state-of-the-art IR models that has been reported to work well on the MS MARCO passage retrieval benchmark [34]. The implementation[2] for our proposed method and the baselines is made available for research purposes.

[2] https://github.com/suchanadatta/WRIG.git

Table 3. Characteristics of the datasets used in our QPP experiments. The suffix 'S70' indicates that documents detected as spam with confidence scores higher than 70% were removed from the collection. 'Avg.$|Q|$' and 'Avg.#Rel' denote the average number of query terms and the average number of relevant documents, respectively. Since the topic identifiers for MS MARCO training set and TREC-DL are in no particular order, the corresponding column is left empty.

Collection (#docs)	Topic Set	Ids	#topics	Avg.$\|Q\|$	Avg.#Rel
Disks 4,5 minus CR	TREC-6	301-350	50	2.54	79.36
(528,155)	TREC-7	351-400	50	2.42	93.48
	TREC-Robust	601-700	100	2.88	37.20
	TREC-8	400-450	50	2.38	94.56
CWeb09B-S70 (29,038,220)	TREC-Web	1-200	200	2.42	16.02
MS MARCO Passage	MS MARCO Train	–	808,731	6.37	1.06
(8,841,823)	TREC-DL'19	–	43	5.40	58.16
	TREC-DL'20	–	54	6.04	30.85

6.2.2 Datasets. We experiment with three standard ad-hoc IR collections, namely the TREC-Robust collection (comprised of news articles), ClueWeb09B [13] (comprised of crawled web pages), and the MS MARCO passage dataset [41] (a question answering dataset that features over $100K$ Bing queries). Table 3 provides an overview of the three datasets. For the ClueWeb09B experiments, we used the Waterloo spam scores [1] to remove documents with spam confidence > 70%. We denote this subset as CWeb09B-S70 in Table 3.

Note that all the experiments involving ColBERT [34] are executed only on MS MARCO dataset. This is because training a large parameter-driven model such as ColBERT is likely to be ineffective on IR test collections with relevance judgments for a small number of queries. Therefore, we do not report results for ColBERT on either of TREC-Robust or ClueWeb09B (corresponding columns in the results tables are left empty).

6.2.3 Train and test splits. The most common setup for QPP experiments in the literature usually involves repeatedly partitioning a set of queries randomly into two parts. The *train set* is used to tune the hyper-parameters for each method under investigation, and the optimal values of these hyper-parameters are then used to evaluate the QPP effectiveness on the *test set* of queries [56, 62, 64]. In our experiments, we also use an identical setup for the TREC-Robust and ClueWeb09B collections, which do not have dedicated training data. Across 30 splits, we randomly generate equally-sized train:test partitions. Each time we train the model on the train split and evaluate the QPP effectiveness on the test split with the optimal setting of hyper-parameters. Finally we report the average outcome obtained for 30 test-folds.

However, for the MS MARCO test collection, since a designated train:test split is available, we tune model hyper-parameters on the training set and report results on the TREC-DL dataset (a subset of the MS MARCO test set) with the optimal parameter setting as prescribed in [3].

It is worth noting that the training set is only used to optimally learn the parameters of a supervised neural model; this set of topics is not used for QPP evaluation. Moreover, since we investigate unsupervised QPP approaches only, the training set of topics has no effect on learning the parameter values (unlike the case of a supervised approach).

6.2.4 Query variants. In addition to using automatically-generated query variants, to allow a direct comparison between WRIG and the additive smoothing technique proposed in [64], we also conducted experiments using the manually-formulated variants of the TREC-Robust queries from

Table 4. Retrieval effectiveness obtained with the statistical model LM-Dir and two different neural re-rankers (DRMM and ColBERT) on the TREC-Robust, ClueWeb09B, and TREC-DL datasets, with $k = 100$ top-retrieved documents. The DRMM parameters, L and M, denote the number of feed-forward layers and the number of quantization intervals, respectively; and m stands for embedding dimension in ColBERT model.

Topic Set	Method	Parameters	MAP
TREC-Robust	LM-Dir	$\mu = 1000$	0.2127
	$\text{DRMM}_{\text{tanh}}$	$L = 1, M = 30$	0.2743
	$\text{DRMM}_{\text{sigmoid}}$	$L = 1, M = 30$	0.2621
ClueWeb09B	LM-Dir	$\mu = 1000$	0.1332
	$\text{DRMM}_{\text{tanh}}$	$L = 1, M = 30$	0.1876
	$\text{DRMM}_{\text{sigmoid}}$	$L = 1, M = 30$	0.1504
TREC-DL	LM-Dir	$\mu = 1000$	0.2954
	$\text{DRMM}_{\text{tanh}}$	$L = 1, M = 30$	0.3206
	$\text{DRMM}_{\text{sigmoid}}$	$L = 1, M = 30$	0.3085
	ColBERT	$m = 128$	0.4189

the UQV dataset [5]. To generate the variants for each TREC query in the UQV dataset, authors in [5] provided a narrative illustrating the information seeking situation to a number of participants, who were then asked to formulate queries and their interactions were logged. The authors of [5] then post-processed those logged queries, e.g., duplicates were removed, spelling errors were corrected etc. Finally, given a manually-created back-story corresponding to a TREC query, they asked participants to formulate appropriate queries.

In our work, for investigating how the number of query variants, $|\mathcal{E}_Q|$, influences the relative effectiveness of an input query, we tried out different values of $|\mathcal{E}_Q|$ from $\{5, 10, 15, 20, 25\}$. We observed that the optimal results were obtained at $|\mathcal{E}_Q| = 10$, both for WRIG and the additive smoothing technique [64].

6.2.5 Retrieval settings. As the initial retrieval model (the output of which is provided as an input to neural re-rankers, DRMM and ColBERT), we employ language modeling with Dirichlet smoothing [65], denoted as LM-Dir. We report results with the value of the hyper-parameter μ set to 1000, as prescribed in [66] on top-retrieved $k = 100$ documents. We conducted a grid search to find the optimal value of k from the set $\{5, 10, 15, 20, 25, 50, 100, 300, 500, 1000\}$, as suggested by [62] and we also obtain the best MAP values with $k = 100$ both for statistical and neural models as reported in Table 4. For all our reported experiments, we measure QPP effectiveness on the top-100 documents.

6.2.6 Neural model hyper-parameters. The hyper-parameters to optimize for DRMM are:

- M, the number of quantization intervals used to discretize the cosine similarity values between the constituent word vector pairs of documents and queries, and
- L, the number of feed-forward layers.

The hyper-parameter M of DRMM was optimized by conducting a grid search in the range 10 to 50. We selected $M = 30$ (as in [26]) because this value yielded the highest MAP on target test collections (see Table 4). The number of hidden layers, L, was also chosen via a grid search over $\{1, 2, \ldots, 5\}$. As prescribed in [26], we used the log-count based histogram coupled with idf weighting as inputs for training the DRMM.

For ColBERT, we do not fine-tune the hyper-parameter m - the dimension of the latent layer on which BERT embedding vectors (768 dimensional) are projected. Instead, as prescribed in the paper [34], we set the dimension of the embedding, m, to 128. Other hyper-parameters of ColBERT were set as suggested by the authors [34]. Specifically, the learning rate was set to $3x10^{-6}$ with a batch size 32 and the upper limit of the number of tokens per query N_q was set to 32.

To show that the neural models were indeed trained in an effective manner in our experimental setup, we report the mean average precision (MAP) values obtained with LM-Dir, DRMM and ColBERT models on the three of the test collections, i.e. TREC-Robust, ClueWeb09B and TREC-DL as in Table 4.

6.2.7 Evaluation metrics. To measure the correlation between predicted and the ground-truth AP values, we employ the standard QPP effectiveness metrics: Pearson's ρ and Kendall's τ. While the former is a value-based correlation, the latter is a rank-based one. Note that we do not report results with Spearman's rank correlation metric because it exhibited similar trends to ρ.

To measure Kendall's τ, the reference or ground-truth ordering of the queries was constructed by sorting the set of the queries in the test-folds by their average precision (AP) values computed with the help of the available relevance judgments. Contrary to other work that reports results with the ground-truth being computed only once for the initial retrieval, our experiments involve two separate ground-truth orderings. The first is for the initial retrieval (LM-Dir) and the second is for the list re-ranked with the neural models.

In addition to the correlation metrics, we also utilize the rank differences of each query to obtain a per query analysis as proposed in [23]. Specifically, for each query the difference (or error in other words) in the rank position assigned by the ground truth AP and that assigned by a QPP method is measured. The authors of [23] named this per query rank error measure as scaled Absolute Rank Error (abbreviated as sARE). Formally speaking, for a given query q of a query set Q, sARE of q with respect to its ground truth AP value is defined as

$$\text{sARE}_{AP}(q_{)} = \frac{|r^p_{-}r^e|}{|Q|}, \tag{17}$$

where r^p and r^e are the ranks assigned to q by the QPP system and the evaluation metric (here, AP), respectively.

6.3 QPP methods investigated

We experiment with a number of standard QPP methods that have been reported to work well in the literature, namely (i) Clarity [14], (ii) WIG [70], (iii) NQC [56, 64], (iv) UEF [54] with NQC as the base estimator denoted as UEF(NQC), and (v) SCNQC [48] (see Section 3.1 for more details on these baseline methods.) In our experiments with UEF as a baseline, we use NQC as the base estimator ϕ (Equation 6) because among all post-retrieval estimator for neural re-rankers, NQC exhibits the maximum correlation as observed in Table 6. As the rank correlation function of UEF(NQC) (Equation 6), we use the Pearson's-ρ as prescribed in [54].

We experimented with two additional baselines, namely i) PFR-QPP [49] (detailed in Section 3.1) and ii) RLS [47]. Recall from Section 3.1 that PFR-QPP in Equation 7 incorporates information both from the initial result list obtained in response to the original query and a second retrieved list produced by the *expanded queries* obtained with RLM. This method thus conducts a QPP on the re-retrieved list of documents.

Our proposed relative difference-based model WRIG, on the other hand, leverages information only from the re-ranked list of documents produced by neural re-rankers (e.g. DRMM or ColBERT). WRIG captures relative information gain through query perturbation from a set of *automatically generated query variants* instead of expanding the original query by RLM. The reason PFR-QPP is employed as a baseline is because it makes use of the re-ranked list of documents as one of the components involved in predicting the performance of the original query.

The working mechanism of RLS, a reference list-based QPP model, is relatively closer to our proposed model WRIG. Both WRIG and RLS make use of the relative information gain from an additional list of equivalent queries and hence RLS serves as a relevant baseline in this paper.

The main difference between WRIG and RLS is that while WRIG generates a set of query variants with similar information needs automatically (as detailed in Section 5), the RLS method on the other hand, augments the original query by adding a *single* term chosen from a distribution of term weights estimated by RLM [36]. The model then makes a decision about the inclusion of each generated variants based on a statistical hypothesis test, the hypothesis being that the means of the two RSV distributions - one for the original query and the other that of the variant, are equal.

It is worth noting in this context that in terms of creating query variants, additive smoothing methodology, i.e JM [64] (detailed in Section 3.2) is, in principle, closer to WRIG than RLS. This is because, as argued in Section 4.4, both WRIG and JM make use of a set of analogous query variants generated either manually (in case of JM) or automatically. Since JM is the closest to WRIG in terms of the working principle, from Table 7 onward, we directly compare the results only between WRIG and JM for ensuring fairness in the comparisons.

Note that we do not include the pre-retrieval QPP approaches, such as AvgIDF or MaxIDF [29, 31] etc. in our empirical investigation because they have been reported to be outperformed by post-retrieval approaches in a number of existing studies [53, 56, 62, 70]. Moreover, since our proposed method is unsupervised, for fair comparisons, we do not consider supervised QPP approaches of [3, 19, 62] as our baselines.

6.4 QPP method hyper-parameters

Most of the baseline predictors that we have reported in this paper involve a number of free parameters to be tuned; we made sure that the results for each method reported uses the optimal parameter settings. For instance, in NQC [56] the free parameter that we tune is the number of top documents (k), used to compute the standard deviation which we choose from $\{5, 10, 15, 20, 25, 50, 100, 300, 500, 1000\}$. In addition to k, SCNQC [48] involves a number of hyper-parameters, namely, α, β and γ as can be seen in Equation 3. We choose the optimal setting of these 3 parameters by a grid search, where $\alpha, \beta, \gamma \in \{0.25, 0.5, 1.0, 1.5, 2.0\}$ as prescribed by [47].

The baseline methods of Clarity [14], UEF [54] (Equations 5 and 6, respectively), the reference list based method - RLS [47], and the pseudo-feedback based PFR-QPP [49] involve estimating a feedback model using the top-m documents. For our experiments, the optimal values of m for each method were obtained with a grid search over the set $m \in \{10, 15, 20, 25, 30, 35, 40, 45, 50\}$.

Both RLS and PFR-QPP include a parameter that indicates number of reference lists L to use in the final prediction which we tune from the set $\{5, 6, \ldots, 15\}$ as suggested by the authors. There is an additional weighting parameter η involved in PFR-QPP (see Equation 7) which is chosen from the set $\{0.1, 0.2, \ldots, 0.9\}$.

6.5 Revisiting the research questions

We now describe the different settings of the QPP methods investigated, as appropriate to the particular research questions.

(a) To investigate **RQ-1**, we apply a relatively simple approach of "stretching out" the RSVs of a neural model to a much larger (theoretically unbounded) interval. More specifically, we apply the $\tanh^{-1}$ and logit, which, respectively, are the inverse of the tanh and sigmoid functions used as the output layers of DRMM.

(b) In relation to **RQ-2**, to find out if the relative difference based approach is better than the additive smoothing of Equation 9, we employ several post-retrieval estimators as the

Table 5. Examples of nomenclature associated with the methods investigated.

Φ	Λ	Q	Description
NQC	$\emptyset$	No-QV	Baseline from [56]
NQC	JM	UQV	Baseline from [64]
NQC	JM	RLM, W2V	Baseline from [64], extended with automatically generated query variants
NQC	WRIG	UQV, RLM, W2V	Our proposed method

underlying estimator within the WRIG model, i.e., we instantiate Φ of Equation 11 with NQC, WIG etc.

(c) To address **RQ-3**, instead of using only an existing set of reference queries to augment a particular estimator (e.g. NQC or Clarity), we tried out two different ways of automatically constructing the set of query variants. The first one among these uses relevance feedback based query term substitution, whereas the second one uses word vector embeddings (see Section 5). We name these two approaches as 'RLM' and 'W2V' in our experiments, respectively.

(d) Next, to investigate **RQ-4**, instead of making use of the variances in the retrieval scores of top-documents, we adopt the more general approach of using the slope of the regressor line as an estimate of QPP (Section 4.5). To distinguish the existing variance-based NQC with the regressor based one, in our experiments we name the former as NQC while the latter is termed as 'LR', e.g., UEF(LR).

6.6 Nomenclature of methods

For the convenience of referring to the QPP methods in our experiments, we adopt the naming convention of identifying a method as a triple of the form $\langle \Phi, \Lambda, Q \rangle$. Each component of a triple is explained as follows:

- Φ is a base QPP estimator, e.g. NQC or WIG.
- $\Lambda \in \{$ WRIG, JM, $\emptyset\}$ indicates whether our proposed method of relative differences (Equation 11), or the existing method of additive smoothing [64] was used to harness information from the query variants ($\emptyset$ corresponds the case of not using any variants).
- $Q \in$ {No-QV, UQV, RLM, W2V} denotes the set of query variants used. More precisely, this set of query variants is either the pre-existing set of queries from the UQV dataset (corresponding to the TREC-Robust set of experiments), or a set of *automatically generated queries* using either of RLM or W2V (section 5.2 and 5.3). 'No-QV' means that no query variations were used.

Note that all method names of the form ⟨*, WRIG, *⟩ *originate as a contribution from this paper.* On the other hand, the names ⟨NQC, JM, *⟩ correspond to the experiments conducted in [64]. See Table 5 for examples.

7 RESULTS

We now present the results of our experiments and the observations made for each QPP method investigated. This is then followed by a detailed analysis of the observed results.

7.1 Main Observations

Table 6 corresponds to the existing baseline approaches. Table 7 investigates how our proposed automatically generated query variants coupled with regression-based estimator - LR, improves the additive smoothing based QPP model - JM. Table 8 presents the main results of our experiments,

Table 6. Comparisons of rank correlation values (measured with Pearson's ρ and Kendall's τ) between statistical model (LM-Dir) and neural models (DRMM and ColBERT) on the 3 different datasets. A post-hoc application of an activation function's inverse is also used to transform the RSV's of DRMM into a wider range. It can be seen that the QPP effectiveness values of neural rankers are considerably lower as compared to the LM-Dir results. Moreover, a post-hoc transformation of the range of the activation functions to $(-\infty, \infty)$ by tanh^{-1} (inverse tanh) or to $[0, \infty)$ by logit (inverse sigmoid) also has a negative impact on QPP effectiveness. PFR-QPP involves reranking of initial retrieved lists which is why we apply these estimators only on neural rerankers (cells for LM-Dir are grayed out). Reported values along the RLS column are to be compared with corresponding WRIG values in Table 8.

		LM-Dir		DRMM$_{\text{tanh}}$		DRMM$_{\text{tanh}^{-1}}$		DRMM$_{\text{sigmoid}}$		DRMM$_{\text{logit}}$		ColBERT	
Dataset	QPP System	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ
Robust	Clarity	0.4863	0.3140	0.3621	0.2618	0.3417	0.2569	0.3314	0.2511	0.3298	0.2523		
	WIG	0.5240	0.4379	0.4010	0.2984	0.3782	0.2903	0.3887	0.2641	0.3753	0.2610		
	NQC	0.5129	0.4331	0.4228	0.3045	0.4100	0.3017	0.4126	0.2775	0.4005	0.2738		
	UEF(NQC)	0.5423	0.4454	0.4517	0.3189	0.4378	0.3120	0.4409	0.2913	0.4196	0.2954		
	SCNQC	**0.5859**	**0.4493**	**0.4831**	**0.3302**	**0.4489**	**0.3136**	**0.4521**	**0.2978**	**0.4201**	**0.2973**		
	PFR-QPP			0.4983	0.3389	0.5024	0.3372	0.4922	0.3074	0.4719	0.3153		
	RLS	0.6219	0.4507	0.5153	0.3682	0.5022	0.3648	0.5198	0.3654	0.4941	0.3507		
CW09B	Clarity	0.2911	0.1841	0.1742	0.1238	0.1730	0.1204	0.1679	0.1224	0.1614	0.1221		
	WIG	0.3492	0.2420	0.2229	0.1547	0.2213	0.1531	0.2187	0.1490	0.2173	0.1425		
	NQC	0.3478	0.2313	0.2293	0.1601	0.2278	0.1589	0.2215	0.1456	0.2190	0.1538		
	UEF(NQC)	0.3562	0.2351	0.2347	0.1612	0.2334	0.1598	0.2246	0.1554	0.2238	0.1543		
	SCNQC	**0.3588**	**0.2463**	**0.2363**	**0.1674**	**0.2358**	**0.1656**	**0.2271**	**0.1578**	**0.2256**	**0.1546**		
	PFR-QPP			0.3019	0.2105	0.2641	0.1988	0.2549	0.1923	0.2511	0.1945		
	RLS	0.4051	0.2685	0.2976	0.2133	0.2519	0.2078	0.2688	0.1974	0.2621	0.2042		
TREC-DL	Clarity	0.2672	0.2206	0.2043	0.1822	0.2035	0.1751	0.2112	0.1853	0.2091	0.1834	0.2314	0.2146
	WIG	0.3973	0.3789	0.2802	0.2300	0.2794	0.2287	0.2788	0.2257	0.2763	0.2248	0.3086	0.2919
	NQC	0.3929	0.3659	0.2774	0.2241	0.2745	0.2212	0.2723	0.2198	0.2717	0.2132	0.3041	0.2848
	UEF(NQC)	0.3991	0.3672	0.2813	0.2315	0.2791	0.2303	0.2806	0.2278	0.2790	0.2245	0.3185	0.2963
	SCNQC	**0.4013**	**0.3689**	**0.2841**	**0.2359**	**0.2822**	**0.2319**	**0.2790**	**0.2326**	**0.2767**	**0.2321**	**0.3192**	**0.2978**
	PFR-QPP			0.3362	0.2601	0.3276	0.2544	0.2842	0.2296	0.2743	0.2221	0.3278	0.3312
	RLS	0.4177	0.3523	0.3553	0.2556	0.3324	0.2579	0.3018	0.2398	0.3043	0.2321	0.3502	0.3354

where we compare the performance of JM based extensions (e.g. ⟨UEF(LR), JM, UQV⟩) to our proposed method WRIG using either existing query variants (UQV) or automatically generated ones (RLM/W2V).

To interpret the results of Table 8, comparisons should be made across each group of results, e.g., the best results on DRMM$_{\text{tanh}}$ with our proposed approach is 0.6524 (see Table 8), whereas the best achievable with the extended baseline of JM is only 0.5281 (i.e. WRIG improves the prediction by about 23.54% over JM). Since results reported for RLS in Table 6 are reasonably related to that of WRIG in Table 8, we repeat the performance of RLS and WRIG in Table 9 for convenience.

Since there exists no manually-generated query variants for the ClueWeb09B and TREC-DL datasets, the corresponding rows are shown as shaded in both Tables 7 and 8. Moreover, since we report results for the TREC-DL dataset with the ColBERT model only (recall from the discussion in Section 6.2.2 that ColBERT is a data-hungry model and requires a large training set, which is not available for the TREC Robust and the Clueweb datasets), the cells corresponding to the DRMM models are also shown shaded. We now enlist the other observations that can be made from the results of the experiments.

Off-the-shelf QPP methods do not work effectively for neural models. This observation is in relation to **RQ-1** and can be observed from Table 6, by comparing the ρ and τ values obtained for LM-Dir vs. the ones obtained for both the neural models. It can be seen that there is a significant

Table 7. QPP results on the 3 individual datasets for the use of the proposed automatically-generated query variants (shown as RLM and W2V in the table), coupled with the proposed regression-based estimator (LR) to improve the effectiveness of the baseline additive smoothing based approach of [64], denoted as JM in the table. As per the nomenclature in Table 5, these results correspond to tuples of the form ⟨UEF(LR), JM, *⟩. The best results in each group are bold-faced.

		TREC-Robust				ClueWeb09B				TREC-DL			
		JM				JM				JM			
		SCNQC		UEF(LR)		SCNQC		UEF(LR)		SCNQC		UEF(LR)	
Model	Variants	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ
LM-Dir	No-QV	0.5859	0.4493	0.5931	0.4467	0.3588	0.2463	0.3604	0.2515	0.4013	0.3689	0.4078	0.3653
	UQV [64]	0.6482	0.4607	0.6583	0.4729								
	RLM	0.6502	0.4891	0.6610	0.4938	0.4072	0.2766	0.4146	0.2874	0.4268	0.3809	0.4311	0.3857
	W2V	0.6717	0.4834	**0.6801**	**0.4973**	0.4186	0.2994	**0.4248**	**0.3083**	0.4219	0.3873	**0.4302**	**0.3896**
DRMM$_{\text{tanh}}$	No-QV	0.4831	0.3302	0.4974	0.3409	0.2363	0.1674	0.2481	0.1735	0.2841	0.2359	0.2924	0.2342
	UQV [64]	0.4426	0.3213	0.4533	0.3341								
	RLM	0.5047	0.3772	0.5172	0.3818	0.2956	0.2132	0.3010	0.2103	0.3404	0.2653	0.3498	0.2714
	W2V	0.5204	0.4089	**0.5281**	**0.4111**	0.3144	0.2289	**0.3302**	**0.2314**	0.3482	0.2907	**0.3502**	**0.3042**
DRMM$_{\text{sigmoid}}$	No-QV	0.4521	0.2978	0.4602	0.3110	0.2271	0.1578	0.2351	0.1649	0.2790	0.2326	0.2865	0.2389
	UQV [64]	0.4303	0.3018	0.4428	0.3082								
	RLM	0.4882	0.3642	0.4921	0.3504	0.2955	0.2043	0.2987	0.2076	0.3240	0.2612	0.3395	0.2633
	W2V	0.5091	0.3987	**0.5118**	**0.4029**	0.3083	0.2038	**0.3076**	**0.2242**	0.3362	0.2811	**0.3431**	**0.2913**
ColBERT	No-QV									0.3192	0.2978	0.3311	0.3004
	UQV [64]												
	RLM									0.3541	0.3278	0.3662	0.3314
	W2V									0.3808	0.3412	**0.3854**	**0.3469**

Table 8. A comparison between the additive smoothing [64] *enhanced with the use of query variants* for a fair comparison with WRIG. The best results from Table 7, i.e., ⟨UEF(LR), JM, *⟩, are repeated here for convenience. Bold-faced numbers denote the best results in each group. The improvements of the best results obtained with WRIG vs. the extended baselines are significant (t-test with 95% confidence).

		TREC-Robust				ClueWeb09B				TREC-DL			
		JM		WRIG		JM		WRIG		JM		WRIG	
Model	Variants	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ
LM-Dir	UQV [64]	0.6583	0.4729	0.6349	0.4590								
	RLM	0.6610	0.4938	0.6558	0.4725	0.4146	0.2874	0.3781	0.2793	0.4311	0.3857	0.4129	0.3622
	W2V	**0.6801**	**0.4973**	0.6732	0.4793	**0.4248**	**0.3083**	0.4092	0.2764	**0.4302**	**0.3896**	0.4223	0.3721
DRMM$_{\text{tanh}}$	UQV [64]	0.4533	0.3341	0.5167	0.3694								
	RLM	0.5172	0.3818	0.6109	0.4493	0.3010	0.2103	0.3709	0.2541	0.3498	0.2714	0.3856	0.3318
	W2V	0.5281	0.4111	**0.6524**	**0.4782**	0.3302	0.2314	**0.4136**	**0.2979**	0.3502	0.3042	**0.4097**	**0.3511**
DRMM$_{\text{sigmoid}}$	UQV [64]	0.4428	0.3082	0.4921	0.3502								
	RLM	0.4921	0.3504	0.5632	0.4202	0.2687	0.1776	0.3490	0.2113	0.3395	0.2633	0.3807	0.3158
	W2V	0.5118	0.4029	**0.6072**	**0.4545**	0.3076	0.2042	**0.3717**	**0.2577**	0.3431	0.2913	**0.3815**	**0.3209**
ColBERT	UQV [64]												
	RLM									0.3662	0.3314	0.4003	0.3784
	W2V									0.3854	0.3469	**0.4317**	**0.3820**

difference between the QPP effectiveness values obtained for LM-Dir and neural re-rankers. This indicates that the simplistic approach of stretching out the range of RSVs does not prove beneficial; in fact, it slightly degrades results (see the numbers that correspond to the rows of DRMM$_{\text{tanh}^{-1}}$ and DRMM$_{\text{logit}}$ in Table 6).

Table 9. Comparisons between RLS and WRIG with W2V variants. WRIG improves the correlation significantly (bold-faced) as compared to RLS (t-test with 95% confidence).

	TREC-Robust				ClueWeb09B				TREC-DL			
	RLS		WRIG		RLS		WRIG		RLS		WRIG	
Model	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ	P-ρ	K-τ
LM-Dir	0.6219	0.4507	**0.6732**	**0.4793**	0.4051	0.2685	**0.4092**	**0.2764**	0.4177	0.3523	**0.4223**	**0.3721**
DRMM$_{\text{tanh}}$	0.5153	0.3682	**0.6524**	**0.4782**	0.2976	0.2133	**0.4136**	**0.2979**	0.3553	0.2554	**0.4097**	**0.3511**
DRMM$_{\text{sigmoid}}$	0.5198	0.3654	**0.6072**	**0.4545**	0.2688	0.1974	**0.3717**	**0.2577**	0.3018	0.2398	**0.3815**	**0.3209**
ColBERT									0.3502	0.3354	**0.4317**	**0.3820**

Improvements with WRIG are higher than those with JM. This observation, evident from the fact that the bold-faced numbers for DRMM$_{\text{tanh}}$ in the Table 8 are better than the results for JM, answers **RQ-2** in the affirmative. An important implication of this observation is that the 'relative differences' method in WRIG is a better way to leverage additional information from the query variants for QPP estimation.

WRIG outperforms reference list-based approach RLS. This observation is in relation to **RQ-2**. Results from Table 9 confirms the fact that the relative gain from the query variants can be captured more effectively by substituting terms estimated by RLM or W2V model in the original query (in WRIG), than augmenting the query by a single term (as in RLS).

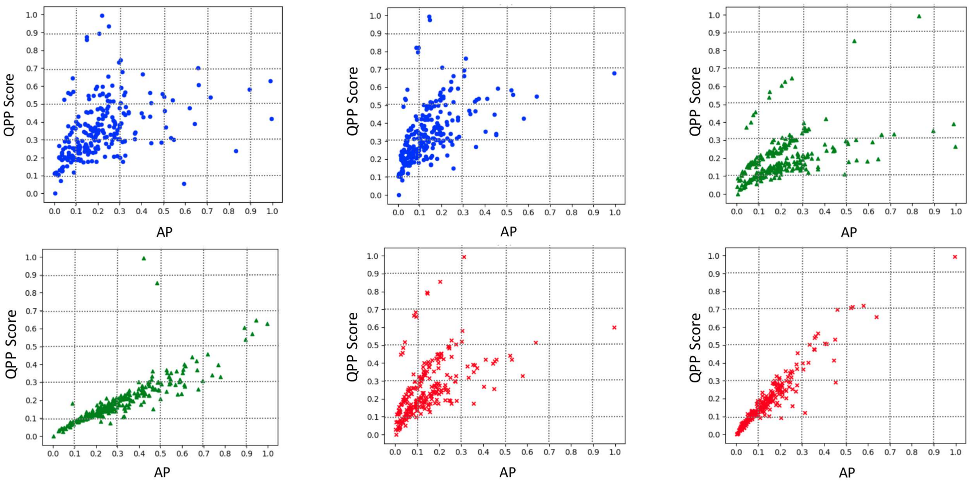
Additive smoothing based augmentation from manually constructed query variants is mostly ineffective for neural models. This is a crucial observation, evident from the drops in the ρ and τ values of (DRMM$_{\text{tanh}}$, UQV) with respect to (LM-Dir, UQV). The implication of this is that the existing smoothing based technique of [64], originally intended to improve QPP effectiveness, contributes to a decrease in QPP effectiveness for neural models. Again, the reason for this is likely attributed to the fact that RSVs (for the original query and its variants) are restricted to a small interval (e.g. $[-1, 1]$ for $\mathtt{tanh}$).

Automatically generated queries improve the performance of the baseline additive smoothing method [64]. This observation relates to the experiments conducted with the additive smoothing based method of [64], to which we feed in the query variants automatically generated by our method presented in Section 5. The purpose of these experiments was to obtain the best possible baseline with additive smoothing against which we could then later compare our proposed method WRIG. It can be seen from Table 7 that [64] works optimally with the presence of *automatically generated queries* – compare UQV rows with 'RLM' and 'W2V' rows in each group for each dataset.

Improvements with automatic variants are higher than those with manual ones. Our proposed way of making use of the information from query variants (Equation 11) produces the most effective results on both the tanh and sigmoid activation functions of DRMM, and also on the sigmoids in ColBERT. This is evident from the WRIG group of results, where correlation values are higher (results with tanh are better in case of DRMM). This observation is related to **RQ-3**, and it demonstrates the following.

Firstly, the automatic generation of query variants yields better results than the manual ones, a likely reason for which is the controlled QPP estimate of the variants (a partial number of query terms from the original query being substituted with other related terms).

Fig. 3. An analysis of the per-query QPP scores for the $\text{DRMM}_{\text{tanh}}$ model for queries in the TREC-Robust dataset. Comparisons are made between the baseline method of additive smoothing with query variants (JM) vs. our proposed way of using relative differences (WRIG). Both the WRIG and JM methods use the best performing base QPP estimate UEF(LR) (Table 8). The order in which results are presented from top-left to bottom-right is as follows: **first row, left**: ⟨UEF(LR), JM, UQV⟩, **first row, middle**: ⟨UEF(LR), WRIG, UQV⟩, **first row, right**: ⟨UEF(LR), JM, RLM⟩, **second row, left**: ⟨UEF(LR), WRIG, RLM⟩, **second row, middle**: ⟨UEF(LR), JM, W2V⟩, and **second row, right**: ⟨UEF(LR), WRIG, W2V⟩. See Table 5 for the naming conventions.



Secondly, we also observe that using the global semantics of word embeddings (W2V) for variant generation is more useful than the local statistics computed from the top-retrieved documents (RLM).

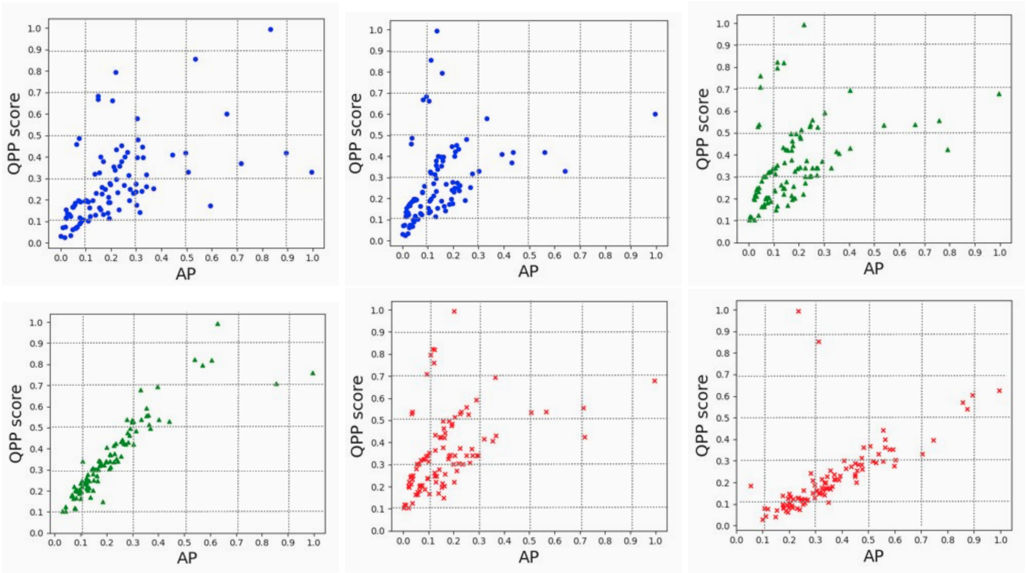
Linear regression outperforms variance-based estimation of QPP. Our proposed methods for estimating the non-uniformity in RSVs outperforms the existing QPP methods. This confirms our hypothesis that existing QPP methods may not be directly effective for neural models when the retrieval scores are strictly bounded within a short interval.

In the context of WRIG, this means that **RQ-4** is answered in affirmative (see in Table 8 that WRIG in combination with the different types of variants, e.g. UQV etc., is particularly beneficial for DRMM). ⟨UEF(LR), WRIG, *⟩ turns out to be the best configuration for WRIG. It is worth mentioning that our proposed regression-based estimator improves additive smoothing based QPP of [64] to a notable extent. Moreover, this observation is also irrespective of manual or automatic query variants as shown in Table 7.

7.2 Analysis

7.2.1 Visualizing the correlations between QPP scores and the retrieval effectiveness. In this section, we present the per-query comparisons between the QPP effectiveness measures obtained with the two methods of leveraging information from the variants, - the baseline JM, and our proposed method WRIG. A convenient way to present the per-query effectiveness results is via a scatter-plot between the normalized values of predicted QPP scores and the true AP values, denoting the predicted and the true query difficulties, respectively.

Fig. 4. Similar to Figure 3, the performance of JM vs. WRIG is shown for the ColBERT re-ranker on the TREC-DL dataset. Results are provided for both manual and automatic generation of query variants, as reported in Table 8. The presentation of results follows the same ordering as in Figure 3, e.g., **top-left**: ⟨UEF(LR), JM, UQV⟩, and so on.

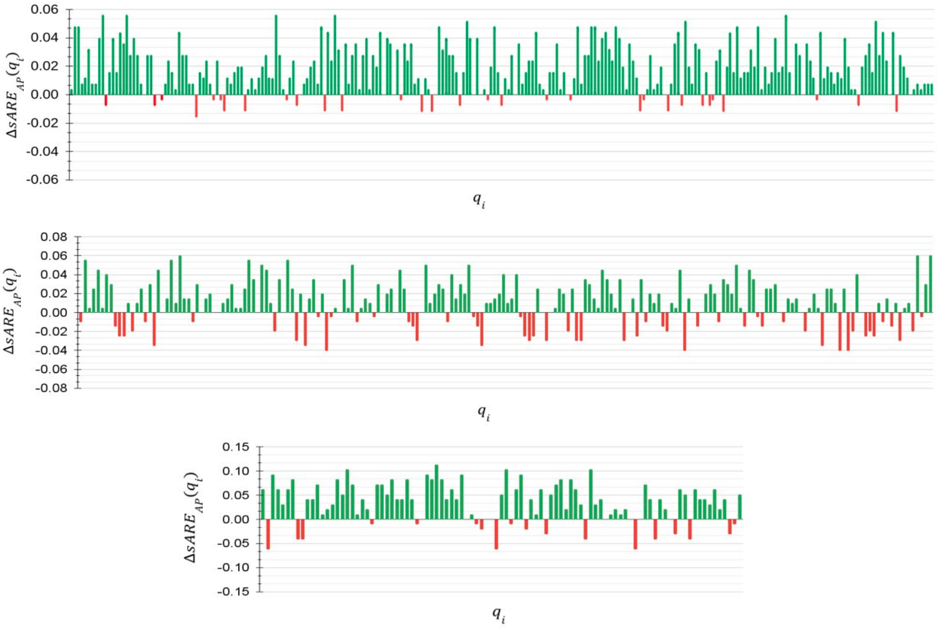


Figures 3 and 4 present the results between the best settings (as per Table 8) obtained with WRIG and JM, i.e., specifically with UEF(LR) as the underlying QPP estimator for both WRIG and JM. Per-query effectiveness measures are shown for two separate combinations of datasets and neural re-rankers, the first being TREC-Robust with DRMM$_{\text{tanh}}$ (Figure 3), and the second being TREC-DL with ColBERT (Figure 4).

A comparison between the adjacent scatter-plots of the same color shows a higher number of outlier points for the plots on the left. This means there is a higher number of cases where the predicted and the true query difficulties do not agree with each other (points away from the left-right diagonal). From Figure 3 and 4, it is observed that the semantic information leveraged with the help of skip-gram word vectors leads to the best results, as evident from the fact that most observations concentrated around the left-right diagonal.

7.2.2 Per-query comparisons of QPP effectiveness. Figure 5 shows a per-query analysis of the QPP effectiveness between the additive smoothing-based JM and our proposed relative difference-based WRIG, in terms of the sARE values (see Equation 17). A convenient way to visualize these differences in ranks, computed respectively by AP values and by QPP scores, is via bar graphs. Each vertical bar in Figure 5 represents the rank error difference for a query with respect to AP values between ⟨UEF(LR), JM, W2V⟩ and ⟨UEF(LR), WRIG, W2V⟩. In other words, we plot the value of $\Delta\text{sARE}_{AP}(q_i) = \text{sARE}_{AP}(q_i; \text{JM}) - \text{sARE}_{AP}(q_i; \text{WRIG})$ for each q_i in the set of queries Q. The green bars indicate that the sARE$_{AP}$ of JM (i.e. the rank error of JM) is higher than that of WRIG. Equivalently, these cases represent those queries for which WRIG outperformed JM, since lower sARE$_{AP}$ values indicate better performance.

Fig. 5. The difference of scaled Absolute Ranked Error with respect to AP values between ⟨UEF(LR), JM, W2V⟩ and ⟨UEF(LR), WRIG, W2V⟩, i.e., $\Delta\text{sARE}_{AP}(q_i) = \text{sARE}_{AP}(q_i;\text{JM}) - \text{sARE}_{AP}(q_i;\text{WRIG})$, for each query q_i. Rank error differences for the first two rows are measured on $\text{DRMM}_{\text{tanh}}$ for TREC-Robust (1^{st} row) and ClueWeb09B (2^{nd} row). The 3^{rd} row shows the difference for ColBERT on the TREC-DL dataset. Note that the green values indicate that the sARE error values for JM are larger, which means that WRIG performs better (smaller error) for these queries. Moreover, the magnitude of the green bars are substantially higher than those of the red ones, which indicates that the relative gains are higher than the losses.



7.2.3 *Relative differences in QPP estimates.* In this section, we conduct an additional analysis on the relative differences between the QPP estimates of an original query and its variants. A high magnitude of relative differences is likely to be more useful to WRIG for QPP. We now investigate if that is indeed the case.

The plots of Figure 6 show that the magnitude of relative differences is fairly large. The dots along a single column correspond to the QPP scores obtained for a query and its variants, the former shown in red, and the latter in green (manual variants) or blue (automatically generated variants). In fact, it is seen that in the case of the manually existing variants of TREC Robust queries (the plot where the QPP estimates of the variants are shown in green), the QPP estimates of some of the queries are higher (potentially these queries being more specific, likely being composed of a higher number of terms), whereas the others are lower. In contrast, we observe that most of the W2V generated variants have higher QPP estimates in comparison to the original queries. It turns out that for DRMM this actually leads to better estimation of the QPP scores (as seen from the higher correlation values in the W2V row as compared to the UQV ones in Table 8).

7.2.4 *Sensitivity to the number of query variants.* We now investigate the effects of parameter choices in the query variant generation process on QPP effectiveness. Similar to the results in Section 7.2.1, we focus on QPP for the neural models $\text{DRMM}_{\text{tanh}}$ and ColBERT, comparing across the additive smoothing (JM) or the WRIG methods of leveraging information from the query variants.

Fig. 6. QPP scores obtained with UEF(LR) - the base estimator in WRIG for each query (both the original and its variants). Vertically aligned points in the plots refer to the QPP scores of the variants (a red point indicates the QPP score of the original query). **Top-left**: DRMM$_{\text{tanh}}$ on TREC Robust with UQV variants; **Top-center**: DRMM$_{\text{tanh}}$ on TREC Robust with W2V variants; **Top-right**: DRMM$_{\text{tanh}}$ on Clueweb with W2V variants; **Bottom-left**: DRMM$_{\text{tanh}}$ on TREC-DL with W2V variants; **Bottom-right**: ColBERT on TREC-DL with W2V variants.

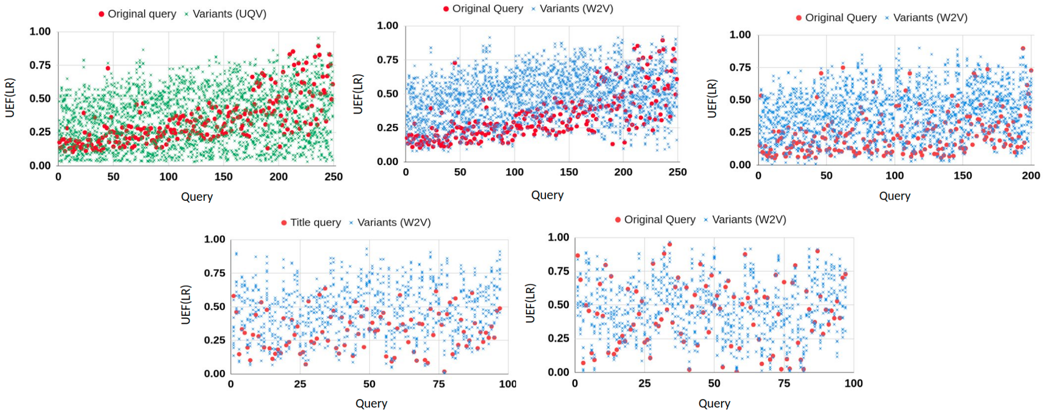


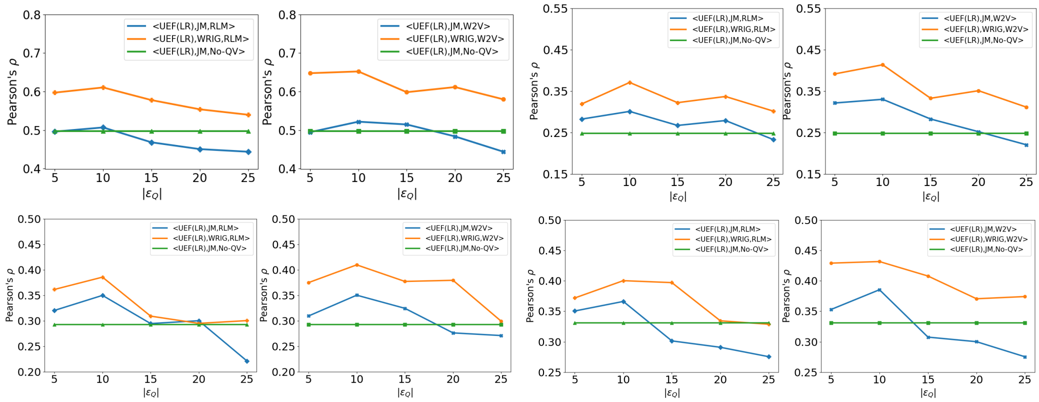
Figure 7 shows that including too few or too many variants does not work well. For both the RLM and the W2V variant generation methods, the optimal results are obtained with 10 query variants. An interesting observation is that the QPP effectiveness of the additive smoothing method is quite sensitive to the number of variants, with results only improving over the baseline method $\langle$UEF(LR), JM, *$\rangle$ for $|\mathcal{E}_Q| = 10$.

8 CONCLUSIONS AND FUTURE WORK

In this paper, we demonstrated that off-the-shelf application of existing query performance prediction (QPP) approaches fail to yield effective results for neural models. This can be attributed to the fact that the retrieval scores obtained from a neural model are restricted within a small interval, e.g. in $[0, 1]$. To improve the QPP estimate for neural models, we propose to use additional information from a set of queries that express a similar information need to the current one (these queries are called variants). The key idea of our proposed method, named Weighted Relative Information Gain (WRIG), is to estimate the performance of these variants, and then to improve the QPP estimate of the original query based on the relative differences with the variants. The hypothesis is that if a query's estimate is significantly higher than the average QPP score of its variants, then the original query itself is assumed (with a higher confidence) to be one for which a retrieval model works well.

Another contribution of the paper is the finding that a linear regression based estimate fitted to the retrieval scores outperforms existing approaches, such as standard deviation [56] or information gain [70] based estimates. Our experiments showed that WRIG outperforms the previously studied way of incorporating information from query variants in the form of additive smoothing [64]. We also reported that automatically generated query variants prove effective (even more effective than manually generated variants) in improving QPP estimates. This indicates that one may not require a set of highly-precise equivalent queries for the purpose of improving QPP estimates on the original queries. We found that among our two proposed ways of generating the query variants - a) via RLM-based and b) via word embedding based term substitutions, the latter performs better.

Fig. 7. Sensitivity of WRIG and JM with respect to the number of variants used to estimate the QPP score for each query. **Row 1, Col 1**: DRMM$_{\text{tanh}}$ on TREC Robust with RLM for variant generation; **Row 1, Col 2**: DRMM$_{\text{tanh}}$ on TREC Robust with W2V for variant generation; **Row 1, Col 3**: DRMM$_{\text{tanh}}$ on Clueweb with RLM for variant generation; **Row 1, Col 4**: DRMM$_{\text{tanh}}$ on TREC Clueweb with W2V for variant generation; **Row 2, Col 1**: DRMM$_{\text{tanh}}$ on TREC-DL with RLM for variant generation; **Row 2, Col 2**: DRMM$_{\text{tanh}}$ on TREC-DL with W2V for variant generation; **Row 1, Col 3**: ColBERT on TREC-DL with RLM for variant generation; **Row 1, Col 4**: ColBERT on TREC-DL with W2V for variant generation.



In future, we plan to leverage the information from query variants in a supervised manner to potentially improve the QPP estimates.

Acknowledgement. This work was partially supported by Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289_P2.

REFERENCES

[1] 2021. Waterloo Spam Rankings for the ClueWeb09 Dataset. https://plg.uwaterloo.ca/~gvcormac/clueweb09spam/. Accessed: 2021-05-25.

[2] Wasi Uddin Ahmad, Kai-Wei Chang, and Hongning Wang. 2019. Context Attentive Document Ranking and Query Suggestion. In *SIGIR*. Association for Computing Machinery, New York, NY, USA, 385–394.

[3] Negar Arabzadeh, Maryam Khodabakhsh, and Ebrahim Bagheri. 2021. BERT-QPP: Contextualized Pre-Trained Transformers for Query Performance Prediction. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*. Association for Computing Machinery, New York, NY, USA, 2857–2861.

[4] Avi Arampatzis and Stephen Robertson. 2011. Modeling Score Distributions in Information Retrieval. *Inf. Retr.* 14, 1 (Feb. 2011), 26–46.

[5] Peter Bailey, Alistair Moffat, Falk Scholer, and Paul Thomas. 2016. UQV100: A Test Collection with Query Variability. In *In Proc. of SIGIR '16*. Association for Computing Machinery, New York, NY, USA, 725–728.

[6] Peter Bailey, Alistair Moffat, Falk Scholer, and Paul Thomas. 2017. Retrieval Consistency in the Presence of Query Variations. In *Proc. of SIGIR '17*. Association for Computing Machinery, New York, NY, USA, 395–404.

[7] NJ Belken, PB Kantor, EA Fox, and JA Shaw. 1995. Combining evidence of multiple query representation for information retrieval. *Information Processing and Management* 31, 3 (1995), 431–448.

[8] Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson. 2008. Selecting Good Expansion Terms for Pseudo-Relevance Feedback. In *SIGIR '08* (Singapore, Singapore). Association for Computing Machinery, New York, NY, USA, 243–250.

[9] David Carmel and Elad Yom-Tov. 2010. Estimating the Query Difficulty for Information Retrieval. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '10)*. Association for Computing Machinery, New York, NY, USA, 911.

[10] David Carmel, Elad Yom-Tov, Adam Darlow, and Dan Pelleg. 2006. What Makes a Query Difficult?. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Seattle,

Washington, USA) *(SIGIR '06)*. Association for Computing Machinery, New York, NY, USA, 390–397.

[11] Ben Carterette, Evangelos Kanoulas, Mark M. Hall, and Paul D. Clough. 2014. Overview of the TREC 2014 Session Track. In *TREC (NIST Special Publication, Vol. 500-308)*. National Institute of Standards and Technology (NIST).

[12] Anirban Chakraborty, Debasis Ganguly, and Owen Conlan. 2020. *Retrievability Based Document Selection for Relevance Feedback with Automatically Generated Query Variants.* Association for Computing Machinery, New York, NY, USA, 125–134.

[13] Charles L. A. Clarke, Nick Craswell, Ian Soboroff, and Gordon V. Cormack. 2010. Overview of the TREC 2010 Web Track. In *TREC (NIST Special Publication, Vol. 500-294)*. National Institute of Standards and Technology (NIST).

[14] Steve Cronen-Townsend, Yun Zhou, and W. Bruce Croft. 2002. Predicting Query Performance. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '02)*. Association for Computing Machinery, New York, NY, USA, 299–306.

[15] Steve Cronen-Townsend, Yun Zhou, and W. Bruce Croft. 2006. Precision Prediction Based on Ranked List Coherence. *Inf. Retr.* 9, 6 (Dec. 2006), 723–755.

[16] Ronan Cummins. 2014. Document Score Distribution Models for Query Performance Inference and Prediction. *ACM Trans. Inf. Syst.* 32, 1, Article 2 (2014), 28 pages.

[17] Ronan Cummins, Joemon Jose, and Colm O'Riordan. 2011. Improved Query Performance Prediction Using Standard Deviation. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*. Association for Computing Machinery, New York, NY, USA, 1089–1090.

[18] Zhuyun Dai, Chenyan Xiong, Jamie Callan, and Zhiyuan Liu. 2018. Convolutional Neural Networks for Soft-Matching N-Grams in Ad-Hoc Search. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM '18)*. Association for Computing Machinery, New York, NY, USA, 126–134.

[19] Suchana Datta, Debasis Ganguly, Derek Greene, and Mandar Mitra. 2022. Deep-QPP: A Pairwise Interaction-based Deep Learning Model for Supervised Query Performance Prediction. In *WSDM*. ACM, 201–209.

[20] Mostafa Dehghani, Hamed Zamani, Aliaksei Severyn, Jaap Kamps, and W. Bruce Croft. 2017. Neural Ranking Models with Weak Supervision. In *Proc. of SIGIR 2017*. Association for Computing Machinery, New York, NY, USA, 65–74.

[21] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. of NAACL'19*. 4171–4186.

[22] Fernando Diaz. 2007. Performance Prediction Using Spatial Autocorrelation. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*. Association for Computing Machinery, New York, NY, USA, 583–590.

[23] Guglielmo Faggioli, Oleg Zendel, J. Shane Culpepper, Nicola Ferro, and Falk Scholer. 2021. An Enhanced Evaluation Framework for Query Performance Prediction. In *Advances in Information Retrieval*. Springer International Publishing, Cham, 115–129.

[24] Henry Feild and James Allan. 2013. Task-aware Query Recommendation. In *Proc. of SIGIR 2013*. Association for Computing Machinery, New York, NY, USA, 83–92.

[25] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep Sparse Rectifier Neural Networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 15)*, Geoffrey Gordon, David Dunson, and Miroslav Dudík (Eds.). PMLR, Fort Lauderdale, FL, USA, 315–323.

[26] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W. Bruce Croft. 2016. A Deep Relevance Matching Model for Ad-Hoc Retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM '16)*. Association for Computing Machinery, New York, NY, USA, 55–64.

[27] Manish Gupta and Michael Bendersky. 2015. Information Retrieval with Verbose Queries. *Found. Trends Inf. Retr.* 9, 3-4 (2015), 91–208.

[28] Claudia Hauff. 2010. Predicting the Effectiveness of Queries and Retrieval Systems. *SIGIR Forum* 44, 1 (Aug. 2010), 88.

[29] Claudia Hauff, Djoerd Hiemstra, and Franciska de Jong. 2008. A Survey of Pre-Retrieval Query Performance Predictors. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM '08)*. Association for Computing Machinery, New York, NY, USA, 1419–1420.

[30] Claudia Hauff, Djoerd Hiemstra, and Franciska de Jong. 2008. A Survey of Pre-Retrieval Query Performance Predictors. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM '08)*. Association for Computing Machinery, New York, NY, USA, 1419–1420.

[31] Ben He and Iadh Ounis. 2004. Inferring Query Performance Using Pre-retrieval Predictors. In *String Processing and Information Retrieval*. Springer Berlin Heidelberg, Berlin, Heidelberg, 43–54.

[32] Nasreen Abdul Jaleel, James Allan, W. Bruce Croft, Fernando Diaz, Leah S. Larkey, Xiaoyan Li, Mark D. Smucker, and Courtney Wade. 2004. UMass at TREC 2004: Novelty and HARD. In *TREC (NIST Special Publication, Vol. 500-261)*. National Institute of Standards and Technology (NIST).

[33] Yongyu Jiang, Peng Zhang, Hui Gao, and Dawei Song. 2020. *A Quantum Interference Inspired Neural Matching Model for Ad-Hoc Retrieval.* Association for Computing Machinery, New York, NY, USA, 19–28.

[34] Omar Khattab and Matei Zaharia. 2020. *ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT.* Association for Computing Machinery, New York, NY, USA, 39–48.

[35] Oren Kurland, Anna Shtok, David Carmel, and Shay Hummel. 2011. A Unified Framework for Post-Retrieval Query-Performance Prediction. In *Proceedings of the Third International Conference on Advances in Information Retrieval Theory (ICTIR'11)*. Springer Berlin Heidelberg, Berlin, Heidelberg, 15–26.

[36] Victor Lavrenko and W. Bruce Croft. 2001. Relevance Based Language Models. In *Proc. of SIGIR '01* (New Orleans, Louisiana, USA). ACM, New York, NY, USA, 120–127.

[37] Ruirui Li, Ben Kao, Bin Bi, Reynold Cheng, and Eric Lo. 2012. DQR: A Probabilistic Approach to Diversified Query Recommendation. In *Proc. of CIKM 2012*. Association for Computing Machinery, New York, NY, USA, 16–25.

[38] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of NIPS 2013*. Curran Associates Inc., Red Hook, NY, USA, 3111–3119.

[39] Bhaskar Mitra, Milad Shokouhi, Filip Radlinski, and Katja Hofmann. 2014. On User Interactions with Query Auto-completion. In *Proceedings of the 37th International ACM SIGIR Conference on Research Development in Information Retrieval.* Association for Computing Machinery, New York, NY, USA, 1055–1058.

[40] Ali Montazeralghaem, Hamed Zamani, and James Allan. 2020. *A Reinforcement Learning Framework for Relevance Feedback.* Association for Computing Machinery, New York, NY, USA, 59–68.

[41] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. *CoRR* abs/1611.09268 (2016).

[42] Joaquín Pérez-Iglesias and Lourdes Araujo. 2010. Standard Deviation as a Query Hardness Estimator. In *String Processing and Information Retrieval*, Edgar Chavez and Stefano Lonardi (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 207–212.

[43] Jay M. Ponte and W. Bruce Croft. 1998. A Language Modeling Approach to Information Retrieval. In *Proc. of SIGIR 1998*. Association for Computing Machinery, New York, NY, USA, 275–281.

[44] Eun Youp Rha, Wei Shi, and Nicholas J. Belkin. 2017. An exploration of reasons for query reformulations. In *Diversity of Engagement: Connecting People and Information in the Physical and Virtual Worlds - Proceedings of the 80th ASIS&T Annual Meeting, ASIST 2017, Washington, DC, USA, October 27 - November 1, 2017 (Proceedings of the Association for Information Science and Technology, Vol. 54).* Wiley, 337–346.

[45] S.E. Robertson, S. Walker, M.M. Beaulieu, M. Gatford, and A. Payne. 1996. Okapi at TREC-4.

[46] S. E. Robertson. 1997. *The Probability Ranking Principle in IR.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 281–286.

[47] Haggai Roitman. 2017. An Enhanced Approach to Query Performance Prediction Using Reference Lists. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Shinjuku, Tokyo, Japan) *(SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, 869–872.

[48] Haggai Roitman. 2019. Normalized Query Commitment Revisited. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 1085–1088.

[49] Haggai Roitman, Shai Erera, and Bar Weiner. 2017. Robust Standard Deviation Estimation for Query Performance Prediction. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval* (Amsterdam, The Netherlands) *(ICTIR '17)*. Association for Computing Machinery, New York, NY, USA, 245–248.

[50] Haggai Roitman and Oren Kurland. 2019. Query Performance Prediction for Pseudo-Feedback-Based Retrieval. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 1261–1264. https://doi.org/10.1145/3331184.3331369

[51] Haggai Roitman and Oren Kurland. 2019. Query Performance Prediction for Pseudo-Feedback-Based Retrieval. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 1261–1264.

[52] Dwaipayan Roy, Debasis Ganguly, Mandar Mitra, and Gareth J.F. Jones. 2016. Word Vector Compositionality Based Relevance Feedback Using Kernel Density Estimation. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM '16)*. Association for Computing Machinery, New York, NY, USA, 1281–1290.

[53] Dwaipayan Roy, Debasis Ganguly, Mandar Mitra, and Gareth J.F. Jones. 2019. Estimating Gaussian mixture models in the local neighbourhood of embedded word vectors for query performance prediction. *Information Processing and Management* 56, 3 (2019), 1026 – 1045.

[54] Anna Shtok, Oren Kurland, and David Carmel. 2010. Using Statistical Decision Theory and Relevance Models for Query-Performance Prediction. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '10)*. Association for Computing Machinery, New York, NY, USA, 259–266.

[55] Anna Shtok, Oren Kurland, and David Carmel. 2016. Query Performance Prediction Using Reference Lists. *ACM Trans. Inf. Syst.* 34, 4, Article 19 (June 2016), 34 pages.

[56] Anna Shtok, Oren Kurland, David Carmel, Fiana Raiber, and Gad Markovits. 2012. Predicting Query Performance by Query-Drift Estimation. *ACM Trans. Inf. Syst.* 30, 2, Article 11 (2012), 35 pages.

[57] Yongquan Tao and Shengli Wu. 2014. Query Performance Prediction By Considering Score Magnitude and Variance Together. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management* (Shanghai, China) *(CIKM '14)*. Association for Computing Machinery, New York, NY, USA, 1891–1894.

[58] Paul Thomas, Falk Scholer, Peter Bailey, and Alistair Moffat. 2017. Tasks, Queries, and Rankers in Pre-Retrieval Performance Prediction. In *Proceedings of the 22nd Australasian Document Computing Symposium (ADCS 2017)*. Association for Computing Machinery, New York, NY, USA, Article 11, 4 pages.

[59] William Webber, Alistair Moffat, and Justin Zobel. 2010. A Similarity Measure for Indefinite Rankings. *ACM Trans. Inf. Syst.* 28, 4, Article 20 (2010), 38 pages.

[60] Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-End Neural Ad-Hoc Ranking with Kernel Pooling. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, 55–64.

[61] Elad Yom-Tov, Shai Fine, David Carmel, and Adam Darlow. 2005. Learning to Estimate Query Difficulty: Including Applications to Missing Content Detection and Distributed Information Retrieval. In *Proc. SIGIR '05*. Association for Computing Machinery, New York, NY, USA, 512–519.

[62] Hamed Zamani, W. Bruce Croft, and J. Shane Culpepper. 2018. Neural Query Performance Prediction Using Weak Supervision from Multiple Signals. In *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '18)*. Association for Computing Machinery, New York, NY, USA, 105–114.

[63] Hamed Zamani, Javid Dadashkarimi, Azadeh Shakery, and W. Bruce Croft. 2016. Pseudo-Relevance Feedback Based on Matrix Factorization. In *Proc. 25th ACM International on Conference on Information and Knowledge Management (CIKM '16)*. Association for Computing Machinery, New York, NY, USA, 1483–1492.

[64] Oleg Zendel, Anna Shtok, Fiana Raiber, Oren Kurland, and J. Shane Culpepper. 2019. Information Needs, Queries, and Query Performance Prediction. In *Proc. of SIGIR '19*. Association for Computing Machinery, New York, NY, USA, 395–404.

[65] Chengxiang Zhai and John Lafferty. 2001. A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '01)*. Association for Computing Machinery, New York, NY, USA, 334–342.

[66] Chengxiang Zhai and John Lafferty. 2004. A Study of Smoothing Methods for Language Models Applied to Information Retrieval. *ACM Trans. Inf. Syst.* 22, 2 (April 2004), 179–214. https://doi.org/10.1145/984321.984322

[67] Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2020. *An Analysis of BERT in Document Ranking*. Association for Computing Machinery, New York, NY, USA, 1941–1944.

[68] Ying Zhao, Falk Scholer, and Yohannes Tsegay. 2008. Effective Pre-Retrieval Query Performance Prediction Using Similarity and Variability Evidence. In *Proc. ECIR'08*. Springer-Verlag, Berlin, Heidelberg, 52–64.

[69] Yun Zhou and W. Bruce Croft. 2006. Ranking Robustness: A Novel Framework to Predict Query Performance. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management (CIKM '06)*. Association for Computing Machinery, New York, NY, USA, 567–574.

[70] Yun Zhou and W. Bruce Croft. 2007. Query Performance Prediction in Web Search Environments. In *Proc. 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*. Association for Computing Machinery, New York, NY, USA, 543–550.