Learning Context-aware Term Importance for Query Performance Prediction

ABBAS SALEMINEZHAD, Toronto Metropolitan University, Canada
NEGAR ARABZADEH, University of California, Berkeley, USA
SOOSAN BEHESHTI, Toronto Metropolitan University, Canada
EBRAHIM BAGHERI, University of Toronto, Canada

Ad hoc retrieval, a cornerstone task in *Information Retrieval (IR)*, aims to rank documents in response to a user's query, often without prior knowledge of the user's specific information need. While transformer-based neural rankers have achieved state-of-the-art performance in ad hoc retrieval, their effectiveness varies significantly across queries. Certain queries—commonly referred to as *hard queries*—remain particularly challenging, highlighting critical gaps in retrieval models. Identifying these hard queries is essential for improving retrieval systems, motivating the task of *Query Performance Prediction (QPP)*, which aims to estimate the effectiveness of a query without requiring access to relevance judgments. In this paper, we propose Context-Aware Query Performance Prediction (`CA-QPP`), a novel post-retrieval QPP method, which builds on the foundations of perturbation-based QPP methods that hypothesize a relationship between query sensitivity to small perturbations and query retrieval effectiveness. Building on this foundation, our approach exposes the given query to perturbations by constructing two query variations: an *effective variation* emphasizing terms that enhance retrieval and an *ineffective variation* accentuating terms that hinder it. By contrasting the retrieval outcomes of these variations using a cross-encoder model, `CA-QPP` captures the interplay of term contributions and predicts the performance for the given query. We evaluate `CA-QPP` on the widely used *MS MARCO* datasets and their associated query sets, including *TREC DL 2019*, *TREC DL 2020*, *DL-Hard*, *TREC DL 2021*, and *TREC DL 2022*, which feature extensive human-labeled relevance judgments. Our experiments demonstrate that `CA-QPP` consistently outperforms traditional and neural-based QPP baselines across standard correlation metrics, including Pearson's ρ, Kendall's τ, and Spearman's ρ. Through a detailed case study, we further illustrate the mechanics of `CA-QPP` and provide empirical evidence for its ability to model the contextual impact of individual query terms, making it a robust framework for query performance prediction.

CCS Concepts: • **Information systems** → **Information retrieval**; **Evaluation of retrieval results**; *Relevance assessment*.

Additional Key Words and Phrases: Query Performance Prediction, Contextualized Pre-trained transformers, Cross-encoder, Bi-encoder

ACM Reference Format:
Abbas Saleminezhad, Negar Arabzadeh, Soosan beheshti, and Ebrahim Bagheri. 2018. Learning Context-aware Term Importance for Query Performance Prediction. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (ACM Transactions on Intelligent Systems and Technology)*. ACM, New York, NY, USA, 26 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

The ad hoc retrieval task is a fundamental problem in Information Retrieval (IR) that involves retrieving a ranked list of documents from a collection in response to a user's query, without any prior knowledge of the user's specific information need [38]. Ad hoc retrieval is the cornerstone of many IR applications, such as web search engines, digital libraries, and recommendation systems, and its effectiveness is essential for meeting diverse and dynamic user requirements [13]. Transformer-based neural rankers, also known as dense retrievers [3, 30], have significantly advanced the performance of ad hoc retrievers in the past few years compared to traditional sparse retrievers [10].

Authors' Contact Information: Abbas Saleminezhad, abbas.saleminezhad@torontomu.ca, Toronto Metropolitan University, Toronto, On, Canada; Negar Arabzadeh, negara@berkeley.edu, University of California, Berkeley, Berkeley, CA, USA; Soosan beheshti, soosan@torontomu.ca, Toronto Metropolitan University, Toronto, On, Canada; Ebrahim Bagheri, ebrahim.bagheri@utoronto.ca, University of Toronto, Toronto, On, Canada.

ACM Transactions on Intelligent Systems and Technology, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06
https://doi.org/XXXXXXX.XXXXXXX

These dense retriever methods capture complex semantic relationships between query terms and document content, allowing for more precise ranking of documents in response to an input query. Their ability to generate contextualized word representations and model hierarchical patterns has established new benchmarks in retrieval effectiveness [32, 58].

While transformer-based neural rankers have significantly improved ad hoc retrieval performance, empirical evidence has shown that not all queries have benefited from this performance improvement equally [2, 36]. For instance, the *MS MARCO Chameleons* study introduced the concept of *Chameleon queries*—subsets of queries that demonstrate high variability in retrieval performance across different retrieval models [8]. These queries expose weaknesses in retrieval systems, as they remain challenging to handle despite advancements in ranking techniques. For example, this study identified a query subset from the MS MARCO development set, comprising hundreds of queries, which exhibits mean average precision (MAP) scores as low as 0.0286 even using more advanced dense retrievers like ColBERT [44]. This highlights that even state-of-the-art methods struggle with these *hard queries*, which often involve ambiguous phrasing, domain-specific knowledge, or multiple facets of intent.

Identifying such hard queries is critical for improving retrieval systems, as it enables targeted strategies like re-ranking, query reformulation, or enhanced model training for specific subspaces [54, 63]. The task dedicated to this is known as *Query Performance Prediction (QPP)*, which aims to estimate the effectiveness of a query's retrieval outcome without access to relevance judgments [13, 69]. QPP methods can be broadly categorized into two main types: *pre-retrieval* and *post-retrieval* approaches. Pre-retrieval methods rely on query-level features, such as query length, term frequency, or collection statistics, to estimate retrieval performance without requiring the results of an actual retrieval process [34]. Such as a recent work [45] that uses language model perplexity scores of query rewrites as a proxy for predicting retrieval performance. These methods are computationally efficient, making them suitable for scenarios requiring real-time predictions. However, they often lack the granularity needed to capture complex interactions between the query and the document collection, limiting their accuracy for more difficult or ambiguous queries. Post-retrieval methods, on the other hand, operate on the results returned by a retrieval system, leveraging retrieval-specific signals such as score distributions, rank positions, or term overlaps between the query and retrieved documents [37, 40]. These methods provide more accurate predictions by directly analyzing the retrieved content and its relevance to the query. However, they come with increased computational costs since they require executing the retrieval process and processing the results.

Our work in this paper is situated within the realm of post-retrieval QPP methods, which leverage retrieval outcomes to estimate query effectiveness. We are specifically inspired by perturbation-based approaches [3, 67] that identify a correlation between a query's robustness to slight perturbations and its retrieval effectiveness. These methods suggest that degrees of change in retrieval outcomes caused by query perturbations can be meaningful indicators of a query's retrieval effectiveness. The underlying hypothesis of these methods is that queries that are robust to perturbations are likely to exhibit higher retrieval effectiveness. Building on this underlying hypothesis from perturbation-based methods, we extend this idea by considering three key premises in our work: *First*, as has been widely documented already [2], there are many different ways through which the same *information need* can be expressed by the users. This reflects the natural variability in how users articulate their search intent, with differences arising from language choice, phrasing, or emphasis on specific aspects of the information need. *Second*, while all these different formulations of the same information need is possible, not all of these query formulations achieve the same level of retrieval effectiveness [8]. Queries that consist of terms that closely align with the vocabulary and context of relevant documents significantly enhance retrieval outcomes, whereas ambiguous or general query terms introduce noise or misalignment. *Finally*, as shown in perturbation-based QPP methods, the contrast between different query formulations of the same information need can provide a structured framework for estimating query performance. In other words, by systematically comparing variations of the same query and their outcomes, it may be possible to estimate a query's retrieval effectiveness.

Building on these premises, our approach introduces a systematic method for estimating query performance by constructing and contrasting deliberate variations of the *original query*. We are particularly interested in building two specific types of query variations: an *effective variation*, which emphasizes terms that enhance retrieval effectiveness, and an *ineffective variation*, which amplifies terms that hinder retrieval effectiveness. These variations provide a structured contrast that allows us to analyze how term-level contributions shape retrieval outcomes. This perspective

aligns with prior work by Zendel et al. [68], who showed that incorporating multiple formulations of a query tied to the same information need can enhance QPP accuracy. To estimate the performance of the *original query*, we analyze the retrieval results of these variations using a *cross-encoder model* that processes each query variation alongside its corresponding retrieved documents. By contrasting the outcomes of the *effective* and *ineffective variations*, the model learns to predict a continuous score representing the query's overall effectiveness. This contrast-based framework, inspired by the principles of *robustness* and *variation sensitivity* [41], provides a structured and context-sensitive way to estimate query performance.

To evaluate the effectiveness of our approach, we conduct experiments using the widely adopted MS MARCO datasets V1 and V2 and their associated query sets [50]: TREC DL 2019 [44], TREC DL 2020 [16], DL-Hard [44] TREC DL 2021[17], and TREC DL 2022[18]. These datasets provide extensive human-labeled relevance judgments, enabling robust evaluation across a diverse range of query difficulties. Our experiments include comparisons with traditional and neural-based baselines, focusing on both statistical and embedding-based QPP methods. We employ standard correlation metrics such as Pearson's ρ, Kendall's τ, and Spearman's ρ to measure the alignment between predicted and actual query performance. Additionally, we analyze the robustness of our approach with respect to key components, such as term weight estimation, query expansion strategies, and aggregation functions. We show that our approach exhibits stable and more effective performance compared to existing state of the art. In summary, the contributions of our paper can be enumerated as follows:

- We propose `CA-QPP`, a method that uses a *contrastive strategy* for estimating query performance. `CA-QPP` generates two deliberate variations for a given input query: one emphasizing terms that enhance retrieval effectiveness and another amplifying terms that hinder it. By contrasting the retrieval outcomes of these variations, our method provides a systematic approach to understand and predict the effectiveness of the original query.
- We introduce a method to generate effective and ineffective variations of the original query by estimating the impact of individual query terms on retrieval performance. By learning a term-weighting method, we assess the contributions of each query term on the query's retrieval effectiveness and construct query variations that reflect opposing ends of retrieval effectiveness.
- We perform extensive experiments on the MS MARCO datasets and their associated query sets, comparing our method against both traditional and neural-based QPP baselines. These evaluations demonstrate the effectiveness and stability of our approach compared to the state of the art.

2 Related Works

Query Performance Prediction has proven effective in the design of complex information seeking systems, particularly in balancing the trade-off between efficiency and effectiveness [64]. A recent overview by Arabzadeh et al. [6, 7] further emphasizes the growing importance of QPP in practical IR scenarios, highlighting its role not only in ad-hoc retrieval but also in emerging applications like conversational and multi-agent search systems. For instance, QPP has been instrumental in query routing, where the system determines whether to employ a more complex and resource-intensive retriever for challenging queries or a lightweight retrieval strategy for easier ones [62]. Additionally, QPP methods have been applied in scenarios such as asking clarifying questions when user intent is predicted to be ambiguous, thereby enhancing system interaction quality [9]. In [28], a framework for QPP in conversational search was introduced, highlighting the need for models and evaluation protocols tailored to multi-turn interactions. Also, the QPP++ 2025 workshop [48] outlined new directions for QPP in the era of large language models, pointing to broader opportunities and challenges.

Depending on when in the retrieval pipeline QPP is conducted, methods are broadly categorized into pre-retrieval and post-retrieval approaches. Pre-retrieval methods operate before document retrieval, relying solely on query and corpus data [1, 64], while post-retrieval methods incorporate additional information from the retrieved documents, alongside query and corpus statistics. Post-retrieval QPP generally achieves higher predictive accuracy due to the availability of richer retrieval signals. Consequently, our work focuses on post-retrieval QPP methods, which have shown to be particularly effective in this domain.

Post-retrieval QPP methods often rely on statistical properties of document scores. For instance, Clarity [20] measures the divergence between the language models of the top-ranked documents and the overall collection, providing an estimate of query difficulty. Similarly, Weighted Information Gain (WIG)[69] assesses the difference between the average scores of top-ranked documents and the collection as a whole. Another class of metrics, such as NQC and SMV [60, 63], focuses on the variance of retrieval scores among the top-retrieved documents. The underlying assumption is that higher variance in these scores indicates a clear distinction between relevant and non-relevant documents, making the query easier to satisfy. Conversely, low variance suggests that all retrieved documents have similar relevance levels, making it harder to differentiate between them [69]. While these methods are computationally efficient, their heavy reliance on the distribution of retrieval scores limits their applicability across different types of retrievers. Variations in score distributions between retrieval models, particularly between sparse and dense retrievers, often render these methods ineffective [30, 61].

Another class of QPP methods focuses on query robustness as a predictor of performance. These methods operate on the premise that more robust queries are likely to perform better. For instance, prior work [3, 69] injects noise into query representations in both lexical and semantic spaces to assess the stability of retrieval results. The degree of overlap between the results retrieved by the original query and the perturbed query serves as an indicator of robustness. Queries with higher overlap are deemed more robust and thus are expected to yield better retrieval performance. In addition, Nascimento et al. [49] introduced a risk-sensitive evaluation framework for QPP, emphasizing the importance of assessing predictor stability across queries and complementing traditional correlation-based evaluations. Our proposed method, `CA-QPP`, also builds on the concept of query robustness but takes a more targeted approach. Instead of introducing random noise into the query, we inject goal-oriented perturbations, specifically generating query variations with intentionally different levels of effectiveness. By comparing the retrieval results of the original query and its perturbed counterparts—where the perturbation represents either an effective or ineffective query—our approach learns to predict query performance. This structured framework enables `CA-QPP` to assess query effectiveness with greater precision, advancing the robustness-based QPP paradigm.

Another group of QPP methods focuses on comparing the retrieved list of results to an ideal reranker. Shtok et al. [60] proposed a reference list-based framework that estimates query performance by comparing the retrieved list to pseudo-effective and pseudo-ineffective result lists. Similarly, Datta et al. [23] propose a framework that predicts query performance by measuring the relative information gain between a query and its automatically generated variants. In [61], the authors propose two frameworks: one comparing the similarity of the retrieved list with a pseudo-relevance feedback (PRF) model, and the other comparing it with a reranked list produced by a strong pairwise ranker such as DuoT5 [52]. These methods are unsupervised and have demonstrated strong performance among unsupervised QPP approaches.

Recent advancements in neural-based post-retrieval QPP methods have demonstrated superior predictive accuracy by leveraging both static and contextualized representations of queries and documents. NeuralQPP [66] utilizes traditional statistical-based QPP metrics as weak signals to train a supervised model using static embeddings. Similarly, NQA-QPP [33] fine-tunes a transformer model by integrating contextual query embeddings, document embeddings, and document-query interactions into a unified framework to predict query performance. BERT-QPP [5] takes a slightly different approach, employing a cross-encoder architecture focused solely on query-document interactions to fine-tune a contextualized transformer model for performance prediction. However, a limitation of BERT-QPP is its inability to handle long texts or multiple passages due to token length restrictions. To address this, QPP BERT-PL [24] proposes a solution by chunking the text into smaller sections and using a sliding window over the top-retrieved documents to overcome token limit constraints. Another variation of BERT-QPP is proposed in [27], where the retrieval scores are integrated into the BERT-QPP framework, combining score-based metrics with neural-based metrics. This hybrid approach has been shown to outperform both individual approaches.Faggioli et al. [29] proposed a unified framework for QPP in neural IR, showing how predictive features can be derived at different stages of the retrieval pipeline. Most recently, Meng et al. [47] introduced QPP-GenRE, a novel framework that estimates query performance by using large language models to generate pseudo-relevance judgments for top-ranked documents, enabling the prediction of various evaluation metrics and improving interpretability through fine-grained relevance modeling.

While these methods have advanced the field, challenges remain in effectively predicting query performance in ad hoc retrieval settings. Our work builds on these efforts by introducing CA-QPP, which combines contextual embeddings with a score-based setup. Recognizing that transformer models are particularly adept at learning term weightings [43], we first identify which terms contribute positively to query performance and which terms detract from it. Instead of directly learning query performance, we decompose the problem into two steps: (1) learning the term importance of individual query terms using contextualized transformer models, and (2) leveraging these term importance weights to predict overall query performance. This additional step, compared to methods that solely fine-tune a transformer model to predict performance, adds an interpretive layer and simplifies the prediction task. We show that this refinement could enhance the accuracy of QPP.

3 Methodology

3.1 Problem Definition

In the context of information retrieval, the Query Performance Prediction (QPP) task aims to estimate the effectiveness of a retrieval method R in addressing the information need behind a given query q, without relying on relevance judgments. For a query q, a retrieval method R produces a ranked list of documents D_q, expressed as $D_q \leftarrow R(q, C)$. Here:

- C represents the document corpus from which documents are retrieved and ranked.
- $R(\cdot)$ is a retrieval function that takes a query q and the corpus C as input and outputs a ranked list of documents D_q, ordered by their relevance to q.

The effectiveness of $R(q, C)$ is determined by its ability to place the most relevant documents at the top of the ranked list D_q. If relevance judgments are available for query q, the quality of the retrieval results is measured using an evaluation function $\mu(q, C, R)$. Common evaluation metrics μ include Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), and normalized Discounted Cumulative Gain (nDCG). In QPP, the goal is to estimate the effectiveness of the retrieved list D_q, denoted as $\hat{\mu}(q, C, R)$, using a QPP method $\phi(D_q, q, C)$ without access to relevance judgments. The predicted performance $\hat{\mu}(q, C, R)$ is then compared to the actual performance $\mu(q, C, R)$ to assess the accuracy of the QPP method across a set of queries. Ultimately, the objective is to evaluate how well $\hat{\mu}(q, C, R)$ approximates $\mu(q, C, R)$, enabling better prediction of retrieval effectiveness for unseen queries.

3.2 Foundations of Our Approach

Our work in this paper is built on three key premises:

(1) *The same information need can be expressed through different query formulations:* In information retrieval, users often express their search intent in multiple ways, reflecting natural variability in language and search behavior [2]. This variability is critical to consider because different formulations of the same information need interact differently with the underlying retrieval system and document collection. For example, the queries "is sinus infection contagious" and "is sinusitis contagious" convey the same user intent but vary in their phrasing and term usage.

(2) *Not all query formulations achieve the same retrieval effectiveness:* Query terms play differing roles in directing the retrieval system toward relevant documents. Terms that align closely with the vocabulary and context of relevant documents can significantly improve retrieval performance, while ambiguous or general terms may lead to irrelevant or suboptimal results [8]. For instance, in the previous example, the query "is sinusitis contagious" outperforms "is sinus infection contagious" on a BM25 ranker because the term *sinusitis* is more specific and semantically aligned with relevant documents. This variability demonstrates that even slight changes in query formulation can have substantial effects on retrieval performance.

(3) *Sensitivity to query perturbations can correlate with query difficulty:* Previous work on perturbation-based query performance prediction has explored how small perturbations to a query can lead to changes in retrieval effectiveness for that query [3, 67]. These methods rely on the idea that less robust queries—those whose performance varies significantly with perturbations—tend to be more difficult, while highly robust queries are generally more effective. The notion of robustness suggests that examining variations of the same query

can reveal how specific changes impact retrieval effectiveness, making it a plausible strategy for estimating performance.

Our work is particularly inspired by the observation that the same information need can be expressed through different query formulations, each exhibiting varying levels of retrieval effectiveness. We examine whether the contrast in retrieval effectiveness between these variations can serve as a reliable indicator of query performance. This approach is motivated by earlier research that has explored the impact of query perturbations as a means to estimate query performance, leveraging robustness to modifications as a proxy for retrieval effectiveness. More specifically in our work, we create two variations of the original query, each designed to exhibit differing levels of retrieval effectiveness—one exhibiting a higher retrieval effectiveness and the other showing less effectiveness. By contrasting these variations and their retrieval outcomes, we systematically investigate whether contrast between query variations can be used to effectively estimate retrieval effectiveness.

3.3 Estimating Query Performance

We are inspired by methods that adopt robustness to query perturbations for estimating query performance, which typically evaluate performance by introducing small modifications to the *original query* and analyzing the retriever's sensitivity to these changes [3, 67]. These methods operate on the assumption that lower robustness (higher sensitivity) to perturbations indicates a more difficult query. Similar to existing perturbation-based methods, which estimate query performance by contrasting the *original query* with a *modified variation*, our approach constructs two deliberate variations of the original query: one that represents a potentially *more effective version* by emphasizing terms likely to enhance retrieval effectiveness and another that represents a potentially *less effective version* by amplifying terms that may hinder retrieval. By contrasting the retrieval outcomes of these two variations, we estimate the performance of the *original query* in a systematic manner.

More specifically our objective is to estimate query performance by contrasting two variations of the initial query: one designed to represent an *effective query* in which focus is given to query terms that enhance retrieval effectiveness, and the other representing an *ineffective query* by emphasizing query terms that hinder retrieval effectiveness. To construct these variations, we first perform a context-specific classification of the terms in the query, categorizing them based on their contributions to retrieval effectiveness. This systematic approach unfolds in three steps:

(1) *Classifying Terms*: For a given input query, we first categorize each of the query terms into one of three types: *promotive terms* that enhance retrieval effectiveness, *demotive terms* that degrade effectiveness, and *neutral terms* that have negligible or mixed impact on retrieval effectiveness. This classification uses a *term-weighting function* that evaluates the influence of each term within the query's retrieval context.
(2) *Creating Query Variations*: Using the classified terms, we generate two contrasting variations of the query. The *promotive-dominant variation* emphasizes promotive terms to simulate a highly effective retrieval scenario, while the *demotive-dominant variation* focuses on demotive terms to reflect a less effective retrieval scenario. These variations are deliberately constructed to reflect opposing ends of the query effectiveness spectrum.
(3) *Predicting Query Performance*: The retrieval outcomes of the two query variations are contrasted using a cross-encoder model, which processes each variation along with its associated retrieved document. By analyzing the differences between the effective and ineffective query variations, the model learns to predict a continuous score representing the performance of the original query.

Based on Figure 1, each step of our proposed `CA-QPP` is detailed in the following.

3.3.1 Step 1: Classifying Query Terms. The first step in `CA-QPP` is to classify query terms based on their context-specific contributions to retrieval effectiveness. We categorize query terms into three types:

- *Promotive terms*: These terms enhance retrieval effectiveness by aligning the query with relevant documents.
- *Demotive terms*: These terms hinder retrieval effectiveness by introducing ambiguity or misalignment with relevant documents.
- *Neutral terms*: These terms have negligible or mixed effects, often playing syntactic or contextually insignificant roles.

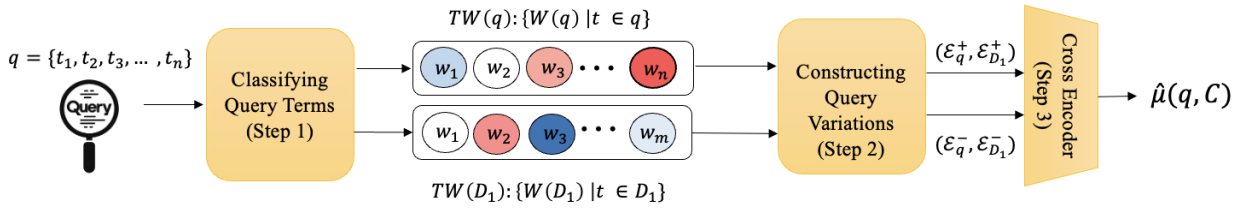


Fig. 1. The overview of our proposed CA-QPP method.

The classification of terms into these three types is achieved using a *term weighting function*, $TW(t_i)$, which assigns a continuous weight to each term t_i in a query q. This weight reflects the term's influence on retrieval performance within its specific context. Promotive terms are assigned positive weights, demotive terms are assigned negative weights, and neutral terms receive weights near zero. The method for learning this function will be detailed later in the paper; however, we note that the term weighting function is both *context-aware* and *performance-driven*; hence, assigns weights to terms depending on the context they appear in and the impact they have on the performance of the query.

3.3.2 Step 2: Constructing Query Variations. Using the classifications of query terms, we construct two query variations from the original query, each of which represents contrasting degrees of retrieval effectiveness:

- The *promotive variation* ($\mathcal{E}_q^+$): This variation amplifies the presence of promotive terms to simulate an effective retrieval scenario. Demotive and neutral terms are included without modification.
- The *demotive variation* ($\mathcal{E}_q^-$): This variation amplifies the presence of demotive terms to simulate an ineffective retrieval scenario. Promotive and neutral terms are included as they are.

In each variation, promotive or demotive terms are amplified through an expansion process where these terms are repeated based on an expansion factor $R(t_i)$, defined as:

$$R(t_i) = \lfloor \alpha \cdot |TW(t_i)| \rfloor, \quad |TW(t_i)| > 0,$$

where $\alpha \in \{10, 20, \dots, 100\}$ is an expansion factor that controls the expansion. Using this factor, the query variations can be formally defined as:

Definition 3.1 (Promotive Expansion ($\mathcal{E}_q^+$)). *The promotive expansion of a query q is:*

$$\mathcal{E}_q^+ = \bigcup_{t_i \in q} \begin{cases} \underbrace{\{t_i, t_i, \dots, t_i\}}_{R(t_i)\ times} & if\, TW(t_i) > 0, \\ \{t_i\} & if\, TW(t_i) \leq 0. \end{cases}$$

Definition 3.2 (Demotive Expansion ($\mathcal{E}_q^-$)). *The demotive expansion of a query q is:*

$$\mathcal{E}_q^- = \bigcup_{t_i \in q} \begin{cases} \underbrace{\{t_i, t_i, \dots, t_i\}}_{R(t_i)\ times} & if\, TW(t_i) < 0, \\ \{t_i\} & if\, TW(t_i) \geq 0. \end{cases}$$

These two variations represent effective and ineffective formulation of the same initial query by only changing the frequency of how promotive and demotive terms appear in each.

3.3.3 Step 3: Contrasting Query Variations with a Cross-Encoder. The final step involves contrasting the two query variations to estimate the performance of the original query. In order to provide *broader context* for each query variation, each variation is appended with a corresponding document that represents the retrieval outcome of that variation. The *promotive variation* ($\mathcal{E}_q^+$) is paired with the expanded version of the relevance judgment document ($\mathcal{E}_{D_{\text{ideal}}}^+$). An expanded version of a document is constructed in the same way as the query. This provides additional

context by indicating that the promotive variation is likely to lead to the retrieval of the relevance judgment document. Conversely, the *demotive variation* ($\mathcal{E}_q^-$) is paired with the expanded version of the top-1 document retrieved by the demotive query ($\mathcal{E}_{D_1}^+$). This indicates that the demotive variation of the query is likely to retrieve less desirable documents represented by the top-1 document for that demotive query.

A cross-encoder model is then trained to compare these two contrasting pairs (i.e., the first pair being $[\mathcal{E}_q^+, \mathcal{E}_{D_{\text{ideal}}}^+]$, and the second pair being $[\mathcal{E}_q^-, \mathcal{E}_{D_1}^-]$, in order to predict a continuous performance score for the original query q:

$$\hat{\mu}(q, C, R) = CE([\mathcal{E}_q^+, \mathcal{E}_{D_{\text{ideal}}}^+], [\mathcal{E}_q^-, \mathcal{E}_{D_1}^-]),$$

where CE is the cross-encoder model. The output $\hat{\mu}(q, C, R)$ is a continuous scalar score representing the predicted performance of the original query. The cross-encoder learns to predict this score by minimizing a cross-entropy loss:

$$L = -\sum_{q \in Q} \left[\mu(q, C, R) \cdot \log \hat{\mu}(q, C) + (1 - \mu(q, C, R)) \cdot \log(1 - \hat{\mu}(q, C))\right],$$

where Q is the set of queries in the training set, $\mu(q, C, R)$ is the true performance and $\hat{\mu}(q, C)$ is the predicted performance for the query q on corpus C.

This contrastive approach enables our method to explicitly learn how individual terms contribute to the retrieval process by analyzing their roles in two opposing retrieval scenarios. By comparing the retrieval outcomes of a promotive-dominant query variation, paired with an ideal relevance judgment document, and a demotive-dominant variation, paired with its top-1 retrieved document, the model is exposed to the full spectrum of how different query terms influence retrieval effectiveness. This structured contrast allows the model to differentiate between terms that align queries with relevant documents and those that degrade retrieval effectiveness by introducing ambiguity or noise. By systematically examining these opposing query variations, the method captures the context-dependent impact of query term composition on retrieval performance, which as we will show in our experiments is both robust and accurate in learning to estimate query performance.

3.4 Context-aware Term Weights

To effectively classify query terms based on their contributions to retrieval performance, we had introduced and adopted a term-weighting function $TW(t_i)$ in Section 3.3. This function assigns continuous weights to individual query terms, reflecting their specific impact within the context of the query and the retrieved documents. In this section, we describe the process for learning this function, including the estimation function and the construction of labeled datasets.

3.4.1 Learning the Weighting Function. To learn the term weighting function, we require a training dataset consisting of labeled query terms, where each term is classified as *promotive*, *demotive*, or *neutral*. For the purpose of explaining how the term weighting function is trained, we assume that the required labeled dataset is already available. The process of constructing this labeled dataset will be detailed in the subsequent section. Within the labeled dataset, term weights $TW(t)$ are provided for each query term of all included queries. The term weights are assigned as follows:

$$TW(t) = \begin{cases} 1 & \text{if } t \in \mathcal{P} \text{ (Promotive term set)}, \\ -1 & \text{if } t \in \mathcal{D} \text{ (Demotive term set)}, \\ 0 & \text{if } t \in \mathcal{N} \text{ (Neutral term set)}. \end{cases}$$

Simply put, *promotive terms* receive a positive weight of 1, *demotive terms* are assigned a negative weight of -1, and neutral terms are associated with 0. The labeled dataset has assigned these weights to each of the query terms by considering the context in which they appear in the query and the impact their presence in the query has on retrieval effectiveness. Using the labeled terms, we train a regression model to predict term weights $\widehat{TW}(t)$ for unseen contexts. The inputs to the regression model are contextual embeddings of terms, which represent the semantic and positional relationships of terms within the query and its retrieved documents. The regression model learns to map these embeddings to continuous term weights, approximating the true weights $TW(t)$. The model is trained to

minimize the *Mean Squared Error (MSE)* loss:

$$\text{MSE Loss} = \sum_{t \in q} \left(TW(t) - \widehat{TW}(t) \right)^2 .$$

To ensure that term weights reflect not only individual term characteristics but also their contextual interactions, the regression model incorporates an attention mechanism. This mechanism allows the model to evaluate how terms influence one another within the query or retrieved document. By attending to important contextual signals, the model produces term weights $\widehat{TW}(t)$ that capture both term-level and context-dependent contributions to retrieval effectiveness. In the next subsection, we explain how the labeled dataset required for this training process is constructed.

3.4.2 Dataset Construction for Learning Term Weights. For the sake of creating the dataset, assume that we are given a specific information need and two queries that express this need: one that achieves perfect retrieval effectiveness (q_p) and another that is ineffective (q_d). Given the perfect retrieval effectiveness of q_p, the top-1 retrieved document for q_p is equivalent to the relevance judgment document (D_{ideal}) for that query. In contrast, for the ineffective query (q_d), the top-1 retrieved document (D_1) is irrelevant to the information need. We use the pairing of q_p with D_{ideal} and q_d with D_1 to provide labels for the query terms, identifying their types as *promotive*, *demotive*, or *neutral*. We propose that the labeling of term types can be achieved using two different strategies, namely `Query-Aware` and `Query-Agnostic`.

The `Query-Aware` strategy examines term occurrences across the effective and ineffective queries and their associated documents. In this strategy, terms are labeled as follows:

- *Promotive Terms*: Present in q_p and D_{ideal} but absent in q_d and D_1.
- *Demotive Terms*: Present in q_d and D_1 but absent in q_p and D_{ideal}.
- *Neutral Terms*: Common across q_p, q_d, D_{ideal}, and D_1, with mixed or negligible impact.

The rationale behind the `Query-Aware` strategy is that it directly incorporates query context, allowing the model to capture how terms behave differently in effective and ineffective queries. By focusing on the interplay between terms in queries and their retrieved documents, this method ensures that term labels reflect their context-specific roles in retrieval effectiveness.

In contrast, in the `Query-Agnostic` strategy, the focus is on term overlaps between retrieved documents and the relevance judgment. In the `Query-Agnostic` strategy, terms are labeled as follows:

- *Promotive Terms*: Shared between D_{ideal} and D_1, contributing positively to alignment.
- *Demotive Terms*: Exclusive to D_1, introducing noise or reducing alignment with D_{ideal}.
- *Neutral Terms*: Exclusive to D_{ideal}, representing missed opportunities for alignment.

The rationale behind the `Query-Agnostic` method is that it emphasizes document alignment as a measure of retrieval effectiveness. By abstracting away from query-specific details, this method provides a more generalizable view of term contributions, focusing on how terms align the retrieved document D_1 with the ideal document D_{ideal}. This approach is particularly useful for analyzing retrieval performance when query context is less critical or unavailable.

The two proposed strategies, `Query-Agnostic` and `Query-Aware`, can produce gold standard labels for query term under the assumption that we are given an information need along with two competing query variations: one effective (q_p) and one ineffective (q_d). However, in practice, creating such competing queries is not straightforward, as we must ensure that both queries seek to address the same information need but vary in their retrieval effectiveness. To address this, we introduce an inverse process of generating queries from a given passage. Specifically, given a passage, we generate a set of queries that seek to retrieve that passage. This approach leverages the fact that all queries generated from the same passage inherently address the same information need, providing a natural basis for constructing competing query pairs. The process unfolds as follows:

(1) *Passage Selection*: We begin with a collection of passages $C = \{p_1, p_2, \ldots, p_m\}$, where each passage $p \in C$ represents a document fragment associated with a specific information need. In our work, these passages come from the MS MARCO dataset [50].

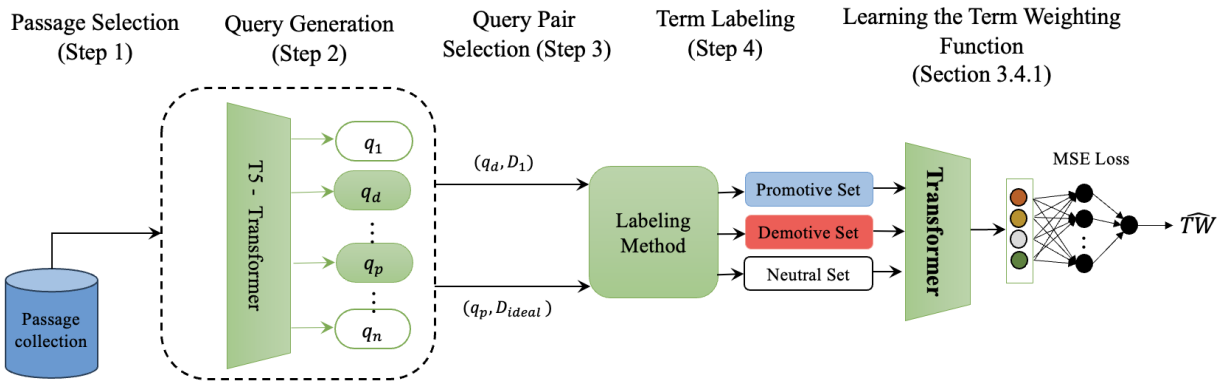


Fig. 2. The proposed process for dataset construction and learning the term weighting function.

(2) *Query Generation*: For each passage p, we use a translation function $\mathcal{T}$ to generate multiple queries $Q_p = \{q_1, q_2, \ldots, q_k\}$ that attempt to retrieve p. The translation function $\mathcal{T}$ can be implemented using a fine-tuned T5 transformer model as already shown in [53]). These generated queries vary in phrasing, term usage, and specificity, reflecting natural variability in user search behavior.

(3) *Query Pair Selection*: From the generated queries Q_p, we identify two competing queries: one effective query (q_p) that achieves near-perfect retrieval effectiveness and one ineffective query (q_d) that performs poorly. The effectiveness of a query is determined by its retrieval performance, measured in terms of metrics such as nDCG@10. The query q_p is paired with its relevance judgment document (D_{ideal}), while q_d is paired with its top-1 retrieved document (D_1).

(4) *Term Labeling*: Once the competing queries and their associated documents are identified, we apply the `Query-Agnostic` and `Query-Aware` strategies to label the terms. These labels serve as gold standard annotations, capturing the types of each term as *promotive*, *demotive*, or *neutral* based on their contributions to retrieval performance.

This process, as outlined in Figure 2, ensures that our dataset consists of competing queries that share the same information need along with appropriate types for each of their terms. This dataset is used for effectively training the term weighting function $TW(t)$.

4 Experimental Setup

4.1 Dataset

We evaluate our proposed approach using two widely adopted benchmark corpora, namely MS MARCO V1 and MS MARCO V2 [50]. The MS MARCO datasets have been extensively utilized for large-scale training and evaluation of various IR and NLP tasks [12, 39]. While MS MARCO V1 is characterized by a smaller collection of passages and queries, MS MARCO V2 expands significantly upon its predecessor, providing a larger and more diverse set of passages, queries, and additional metadata such as document titles and URLs. By incorporating both versions in our experiments, we aim to demonstrate the robustness and generalizability of our method across datasets of varying complexity and scale. We further evaluate our approach on three query sets associated with the MS MARCO V1 passage collection and two query sets associated with the MS MARCO V2 passage collection. Each of these query sets has been comprehensively annotated using a four-level graded relevance scale:

MS MARCO V1 Passage Collection: This collection comprises 8.8 million passages extracted from real web documents, paired with over 500,000 real anonymized Bing queries. We utilized the TREC Deep Learning Tracks from 2019 and 2020, as well as the DL-Hard dataset:

- **TREC DL 2019 (DL-2019)** [19]: A set of 43 queries from the TREC Deep Learning Track 2019.
- **TREC DL 2020 (DL-2020)** [16]: A set of 54 queries from the TREC Deep Learning Track 2020.

- **DL-Hard** [44]: A set of 50 most challenging, unlabeled, and poorly performing queries from TREC 2019 and 2020. This dataset includes higher number of relevance judgments per query compared to DL-2019 and DL-2020 datasets, focusing on those that are harder to satisfy. This focus is particularly important for the QPP task, as it is crucial to accurately predict queries that are more likely to fail.

MS MARCO V2 Passage Collection: A corpus containing 138 million passages derived from approximately 11.9 million documents. The passages were generated using a query-independent algorithm, enhancing the diversity and scale of the dataset. For evaluation, we utilized the TREC Deep Learning Tracks from 2021 and 2022. We utilized the TREC Deep Learning Tracks from 2019 and 2020, as well as the DL-Hard dataset:

- **TREC DL 2021 (DL-2021)** [17]: This dataset includes 53 queries from the TREC Deep Learning Track 2021, specifically designed to evaluate passage retrieval methods. Queries were selected to represent realistic search tasks and are annotated with relevance judgments gathered through a pooling strategy, ensuring high-quality, reusable test collections.
- **TREC DL 2022 (DL-2022)** [18]: This dataset comprises 76 queries from the TREC Deep Learning Track 2022. The queries were carefully curated to challenge retrieval systems by focusing on more complex information needs. Extensive pooling was used to generate robust relevance judgments, aiming to enhance the long-term reusability and comparability of the evaluation data.

These datasets were selected for their extensive human-labeled relevance judgments per query, providing a robust basis for evaluating our approach's performance. They have all been used in many QPP baselines [5, 27, 30, 57].

4.2 Aggregation Strategy

For each query q, performance predictions are derived using the two labeling methods described in Section 3.4. The model outputs a predicted performance score for each query in the test set:

$$\mathcal{L}_{\hat{\mu}_q} = \{\hat{\mu}(q) \mid q \in Q_{\mathcal{T}}\},$$

where $Q_{\mathcal{T}}$ is the set of all test queries, and $\mathcal{L}_{\hat{\mu}_q}$ represents the predicted scores for q under a specific labeling method. To combine the performance scores from the two labeling methods into a single prediction, we define an aggregation function g, which takes the score lists $\mathcal{L}_{Q-aware_q}$ and $\mathcal{L}_{Q-agnostic_q}$ and produces the final aggregated predicted performance. The $\mathcal{L}_{Q-aware_q}$ and $\mathcal{L}_{Q-agnostic_q}$ are the predicted list of queries under `Query-Aware` and `Query-Agnostic` labeling strategy respectively. The following aggregation functions are considered, each with a practical rationale:

- *Maximum Value*: Selects the higher of the two scores. This approach is conservative, assuming that the best labeling method provides the most accurate reflection of query performance. It is useful when overestimating performance is less risky than underestimating it.
- *Minimum Value*: Selects the lower of the two scores. This method is risk-averse, ensuring that the most challenging scenario (as judged by either labeling method) dominates. It is appropriate when avoiding overly optimistic predictions is critical.
- *Mean Value*: Computes the average of the two scores. This balanced approach assumes both labeling methods contribute equally to the final prediction, smoothing out any biases introduced by a single method.
- *Reciprocal Rank Fusion (RRF)* [15]: This fusion method combines the score lists $\mathcal{L}_{Q-aware_q}$ and $\mathcal{L}_{Q-agnostic_q}$ using rank-based weighting. RRF emphasizes highly ranked scores from both lists, making it effective when leveraging complementary strengths of the two labeling methods.

The aggregated score list $\mathcal{L}_g(\mu_q)$ serves as the final performance prediction. By integrating predictions from both labeling perspectives, the aggregation strategy ensures the model considers both `Query-Agnostic` and `Query-Aware` approaches in its final performance prediction.

4.3 Evaluation

A common way for evaluating the QPP task is by measuring the correlation between the predicted and actual query performance on a set of queries. Given two lists of query performances—the actual performance and the predicted

performance—the correlation between these lists quantifies the quality of the prediction. A higher correlation indicates a more accurate prediction of retrieval effectiveness [13]. As such, we report the most commonly used linear and rank-based correlation metrics used earlier in QPP baselines [46]. Pearson's ρ is a linear correlation metric that measures the degree of the linear relationship between the predicted and actual query performance scores. Kendall's τ and Spearman's ρ are rank-based correlation metrics that quantify the similarity between the orderings of the queries when ranked by their actual and predicted performance. We note that, from this point forward, we refer to Pearson's ρ as p-ρ, Kendall's τ as k-τ, and Spearman's ρ as s-ρ throughout the paper. To determine statistical significance, we computed correlations using scipy.stats library, which returns both the correlation coefficient and the corresponding p-value. All reported correlations achieved statistical significance with $p < 0.05$, indicating a probability of less than 5% that these correlations occurred by chance under the null hypothesis.

For all query sets introduced in Section 4.1, we predict the actual performance of two widely used IR models: the BM25 ranker, implemented using Pyserini [42], and a dense retriever built on a pre-trained SBERT model[1]. In our setup, SBERT is used as a bi-encoder to independently encode queries and documents into dense vectors, and retrieval is performed via cosine similarity using the FAISS [2] library [4**?**]. The effectiveness of these IR models is evaluated using the official metric for the aforementioned datasets, i.e., nDCG@10.

4.4 Implementation Details

In this section, we explain the details of the training and inference phases of `CA-QPP`.

Dataset for Weight Estimation. In order to generate the required queries in Section refsec::classifying-terms, we adopt the method in [51] where we fine-tune the T5 transformer using its default settings to develop the translation function $\mathcal{T}$ [53]. Using $\mathcal{T}$, we generate queries for passages from the MS MARCO passage collection. Additionally, we focus on generated queries with performance below the threshold of 0.25, considering these as hard queries. We further retain queries that have a perfect retrieval with a performance of 1, which denote easy queries. This ensures the dataset captures the distinction between hard and easy queries. The final dataset contains 151,652 query pairs constructed on top of the MS MARCO V1 collection. This dataset is used to generate labeled data for the regression model, enabling the prediction of term weights.

Term Weight Estimation Model. We adopted the BERT-base-uncased architecture [25] and fine-tuned it for the term weight prediction task as a regression problem. The model was trained for 12 epochs with a learning rate of 2×10^{-5}. The maximum input length was set to 95, and the batch size was set to 16.

Performance Prediction Model. We utilized a cross-encoder architecture implemented with the SentenceTransformer library [55]. This architecture was trained for two epochs on the input pairs generated using the fine-tuned T5 transformer ($\mathcal{T}$), with a batch size of 16. We experimented with three pre-trained LLMs for the cross-encoder, namely `ms-marco-MiniLM-L-12-v2`[3] [65], `bert-base-uncased`[4] [26], and `deberta-v3-base`[5] [35], whose performances we report in the experiments section.

Expansion Function Parameters. To weigh the terms based on their assigned weights, we adopted the mechanism described in [22]. The α scaling factor was used to scale the term weights during the expansion process. We conducted an analysis to evaluate the impact of α on the performance of our proposed approach.

Codebase. We note that for reproducibility purposes, our code and data are publicly available at https://github.com/Saleminezhad/CA-QPP

4.5 Baselines

We evaluate our post-retrieval QPP model against established post-retrieval baselines. These baselines are categorized into traditional and neural-embedding-based approaches. We note that, following previous work [5, 27], if the QPP method has any hyperparameters, such as the cutoff on which the prediction is estimated, we have tuned the results for DL-2020 and DL-Hard on DL-2019. Conversely, we have tuned the results for DL-2019 on DL-2020.

[1]https://huggingface.co/sentence-transformers/msmarco-distilbert-base-tas-b
[2]https://github.com/UKPLab/sentence-transformers/blob/master/docs/pretrained-models/msmarco-v3.md
[3]https://huggingface.co/cross-encoder/ms-marco-MiniLM-L-12-v2
[4]https://huggingface.co/google-bert/bert-base-uncased
[5]https://huggingface.co/microsoft/deberta-v3-base

Table 1. Comparison of CA-QPP Trained on MS MARCO V1 vs state-of-the-art baselines methods across datasets TREC DL 2019, TREC DL 2020, and DL-Hard in terms of $p-\rho$, $k-\tau$ and $s-\rho$. The IR model here is BM25. The highest value in each column is in bold.

	TREC DL 2019			TREC DL 2020			DL-Hard			TREC DL 2021			TREC DL 2022		
	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$
Clarity [20]	0.271	0.229	0.332	0.360	0.215	0.296	0.149	0.099	0.126	0.243	0.191	0.283	0.141	0.069	0.108
WIG [69]	0.310	0.155	0.226	0.204	0.117	0.166	0.331	0.260	0.348	0.162	0.140	0.203	0.248	0.198	0.299
QF [69]	0.295	0.240	0.340	0.358	0.266	0.366	0.210	0.164	0.217	0.05	0.052	0.076	0.217	0.152	0.226
n(σ_x) [21]	0.371	0.256	0.377	0.429	0.298	0.478	0.195	0.120	0.147	0.298	0.258	0.372	0.142	0.196	0.274
RSD [56]	0.460	0.262	0.394	0.426	0.364	0.508	0.362	0.322	0.469	0.250	0.144	0.204	0.029	0.105	0.157
SMV [63]	0.495	0.289	0.404	0.424	**0.391**	**0.539**	0.375	0.269	0.408	0.252	0.192	0.278	0.330	0.157	0.232
NQC [60]	0.466	0.267	0.399	0.464	0.294	0.423	0.384	0.288	0.417	0.271	0.201	0.292	0.337	0.157	0.235
UEF^{NQC} [59]	0.507	0.293	0.432	0.470	0.365	0.482	0.359	0.319	0.463	0.274	0.252	0.385	0.311	0.274	0.383
NeuralQPP [66]	0.289	0.159	0.224	0.283	0.163	0.259	0.173	0.111	0.134	0.177	0.143	0.198	0.159	0.090	0.130
NQA-QPP [33]	0.348	0.164	0.255	0.350	0.125	0.307	0.386	0.297	0.418	0.250	0.223	0.323	0.256	0.213	0.314
BERT-QPP [5]	0.491	0.289	0.410	0.467	0.364	0.448	0.404	0.345	0.472	0.254	0.207	0.302	0.294	0.229	0.316
qpp-PRP [61]	0.321	0.181	0.229	0.189	0.157	0.229	0.090	0.061	0.063	0.004	0.009	0.002	0.136	0.069	0.081
NN-QPP [27]	0.519	0.318	0.459	0.462	0.318	0.448	0.434	0.412	0.508	0.217	0.171	0.226	0.307	0.229	0.316
CA-QPP	**0.583**	**0.377**	**0.541**	**0.531**	0.334	0.471	**0.545**	**0.452**	**0.600**	**0.379**	**0.297**	**0.412**	**0.430**	**0.279**	**0.391**

4.5.1 Traditional Baselines. Traditional baselines rely on the statistical features of the retrieved documents and the query. For example, Clarity [20] measures the KL-divergence between the language models of the retrieved documents and the entire collection. Other methods, such as WIG [69], NQC [60], $n(\sigma\%)$ [21], RSD [56], and SMV [63], use retrieval score statistics to predict query performance. These methods assume the distribution of the relevance score among top-k retrieved documents can be an indicator of query effectiveness. For instance, NQC predicts better performance for queries where the standard deviation of top-ranked retrieval scores is high, as this suggests clear separation between relevant and non-relevant documents. The Utility Estimation Framework (UEF^{NQC}) [59] builds on these ideas in addition to Psuedo-Relevance Feedback (PRF) to improve predictions.

4.5.2 Neural-Embedding-Based Baselines. NeuralQPP [66] was one of the first methods to use unsupervised QPP scores as weak signals to train a supervised model. Another method, NQA-QPP [33] uses a BERT model to learn representations of queries and documents, combining score distributions, query features, and query-document interactions. Similarly, BERT-QPP [5] fine-tunes BERT to predict query retrieval scores directly. Building on BERT-QPP, more recent approaches, such as qpp-BERT-PL [24], use both pointwise training (focusing on individual queries) and listwise training (using top-ranked pseudo-relevant documents) to tackle QPP problem. Another recent method, QPP-PRP [61], evaluates query performance by comparing the ranked list generated by a neural ranker to a re-ranked list produced by a pairwise neural reranker like DuoT5 [52].

5 Results and Findings

In this section, we compare the performance of our method with state-of-the-art baselines. Additionally, we investigate the impact of different components of our approach and study how robust our method is with respect to various choices of hyperparameters and backbone language models. Specifically, we aim to answer the following research questions:

- **RQ1.** How does CA-QPP perform compared to the state-of-the-art traditional post-retrieval QPP and neural-based QPP baselines?
- **RQ2.** How robust is CA-QPP with respect to the choice of the backbone language model?
- **RQ3.** What is the impact of each of the expansion method components, i.e., Query-Agnostic and Query-Aware? and How robust is CA-QPP with respect to experimental weighting functions?
- **RQ4.** What is the impact of the choice of aggregation function on the Query-Agnostic and Query-Aware?

In the following subsections, we explore and answer each of these research questions.

Table 2. Statistical significance test results comparing the CA-QPP approach against selected baseline methods.

Method	TREC DL 2019	TREC DL 2020
SMV	2.15×10^{-3}	6.61×10^{-3}
NN-QPP	2.00×10^{-3}	3.39×10^{-4}

5.1 Comparison With Baselines

In this section, we compare the performance of CA-QPP with state-of-the-art baselines, as summarized in Table 1. The reported results for CA-QPP use DeBERTa as the backbone language model, RRF as the aggregation function, and an expansion factor of $\alpha = 50$. In the following sections, we examine each of these parameters individually and analyze the robustness of CA-QPP with respect to each one.

As seen in the table, CA-QPP consistently outperforms all baselines across most datasets and metrics. Specifically, CA-QPP achieves the highest performance across all correlation metrics ($p - \rho$, $k - \tau$, and $s - \rho$), showing significant improvements over both traditional and neural-based baselines on TREC DL 2019, TREC DL 2021, TREC DL 2022, and DL-Hard. For TREC DL 2020, while CA-QPP outperforms all baselines in linear correlation ($p - \rho$), SMV achieves slightly better results for rank-based correlations ($k - \tau$ and $s - \rho$). However, the margin by which CA-QPP leads on other datasets highlights its overall robustness.

On DL-Hard, CA-QPP demonstrates exceptional performance, surpassing all baselines by a large margin. Many baselines, such as qpp-PRP and n(σ_x), exhibit a significant performance drop on DL-Hard. This dataset is particularly important as it contains challenging queries with practical real-world applications, where accurately predicting query performance is critical for tasks such as query reformulation and query routing.

Looking at the baselines, among the traditional ones, we observe that while score-based methods such as NQC and SMV perform reasonably well on easier datasets, their performance diminishes on DL-Hard and varies considerably on TREC DL 2021 and 2022, showcasing their limitations in handling diverse and challenging scenarios. Neural models, such as BERT-QPP and NN-QPP, show better consistency across datasets compared to traditional baselines. However, they still fall short of CA-QPP, particularly on TREC DL 2019 and DL-Hard.

One of CA-QPP's key advantages is its stability. Unlike baselines such as qpp-PRP, which performs well on TREC DL 2019 but poorly on DL-Hard and newer datasets, CA-QPP maintains high performance across all datasets, including the more recent TREC DL 2021 and 2022.

In response to **RQ1**, the results demonstrated in Table 1 show that CA-QPP not only achieves the best overall performance but also addresses the challenges posed by predicting the performance of difficult queries. Its consistency across datasets and metrics positions it as a robust and reliable approach for post-retrieval QPP.

Figure 3 presents the per-query differences in scaled Absolute Ranked Error (ΔsARE) between our proposed CA-QPP method and two strong baseline approaches: SMV, a representative score-based model, and NN-QPP, a leading neural-based model [31]. The ΔsARE for each query q_i is defined as

$$\Delta\text{sARE}_{AP}(q_i) = \text{sARE}_{AP}(q_i; \text{Baseline}) - \text{sARE}_{AP}(q_i; \text{CA-QPP})$$

where ΔsARE_{AP} quantifies how accurately a model ranks query performance relative to actual AP-based rankings. Positive values of ΔsARE indicate that CA-QPP achieves lower ranking error for the given query, reflecting better predictive performance. As shown, CA-QPP consistently outperforms both baselines across the TREC DL 2019 and TREC DL 2020 datasets, with most queries exhibiting positive ΔsARE values. Moreover, the positive bars are both more frequent and generally larger in magnitude than the negative ones, demonstrating that CA-QPP not only surpasses these baselines on a greater number of queries but also achieves more substantial reductions in ranking error where it does outperform.

We also performed paired t-tests on the per-query sARE (scaled Absolute Rank Error) values to statistically compare our proposed method with selected baseline methods. This approach uses the distribution of per-query errors rather than summary correlation scores, making it suitable for hypothesis testing. To account for multiple comparisons, we applied the Bonferroni correction. The reported p-values are compared against a Bonferroni-corrected significance threshold to determine statistical significance. Table 2 presents the results of these comparisons.

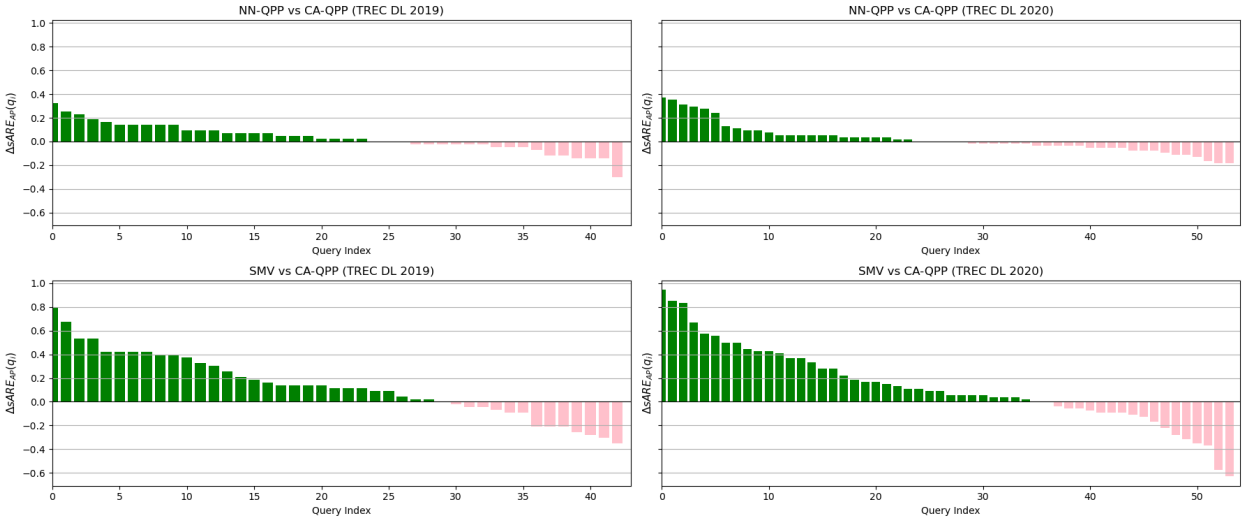


Fig. 3. Per-query differences in scaled Absolute Ranked Error (sARE) with respect to AP values, defined as $\Delta\text{sARE}_{\text{AP}}(q_i) = \text{sARE}_{\text{AP}}(q_i; \text{NN-QPP}) - \text{sARE}_{\text{AP}}(q_i; \text{Proposed})$, are shown for each query q_i. Rows present results on the TREC-DL datasets. Green bars indicate that the baseline method yields higher error than the proposed method (i.e., the proposed method performs better), while red bars indicate the opposite. Notably, the green bars are often taller than the red ones, suggesting that the performance improvements outweigh the degradations in both frequency and magnitude.

5.2 The Impact of Different Types of Information Retrieval Systems

In this section, we examine how our proposed approach performs across different underlying retrieval models. While the main results in Table 1 were based on BM25, a widely used sparse retrieval method, Table 3 presents a similar comparative analysis using the SBERT dense retriever. This allows us to assess the generalizability of CA-QPP when applied to fundamentally different IR systems.

As shown in Table 3, CA-QPP consistently outperforms state-of-the-art baselines across most datasets and metrics even when paired with the dense SBERT retriever. Specifically, CA-QPP achieves the highest correlation scores across all three metrics ($p-\rho$, $k-\tau$, and $s-\rho$) on TREC DL 2019, DL-Hard, and TREC DL 2021, indicating its robustness in handling both typical and challenging query sets. On TREC DL 2020, while BERT-QPP attains slightly higher correlations, particularly in $p-\rho$, $k-\tau$, and $s-\rho$, CA-QPP still demonstrates competitive performance, outperforming most other traditional and neural baselines. Similarly, for TREC DL 2022, although qpp-PRP shows the highest $s-\rho$, our method maintains leading performance on $p-\rho$ and $k-\tau$, underscoring its stable predictive capacity.

When comparing to traditional score-based methods such as NQC and SMV, we observe that while these approaches can perform adequately on simpler query collections, they struggle considerably on more complex collections such as DL-Hard. Neural baselines such as BERT-QPP and NN-QPP generally exhibit improved consistency across datasets but still do not match the comprehensive performance of CA-QPP, especially on harder queries.

Taken together with our earlier results on BM25, these findings illustrate that CA-QPP performs effectively across both sparse and dense retrieval settings. This highlights its versatility and underscores its value as a robust approach for post-retrieval QPP, independent of the specific type of IR system employed.

5.3 The Impact of the Choice of Language Model

In this section, we study the impact of different backbone language models on the performance of CA-QPP. To this end, we replicated our method using three pre-trained language models BERT [25], DeBERTa [35], and MiniLM [65]. These models vary in their architecture, training strategies, and size. We use the BERT-base-uncased version, which is a bidirectional transformer pre-trained on masked language modeling. Additionally, we use the DeBERTa-v3-base model, which introduces disentangled attention mechanisms and enhanced pre-training strategies, aiming to improve

Table 3. Comparison of CA-QPP Trained on MS MARCO V1 vs state-of-the-art baselines methods across datasets TREC DL 2019, TREC DL 2020, and DL-Hard in terms of $p-\rho$, $k-\tau$, and $s-\rho$. The IR model here is the SBERT dense retriever. The highest value in each column is in bold.

	TREC DL 2019			TREC DL 2020			DL-Hard			TREC DL 2021			TREC DL 2022		
	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$
Clarity [20]	0.165	0.132	0.193	0.009	0.022	0.026	0.230	0.146	0.226	0.120	0.160	0.221	0.037	0.145	0.209
WIG [69]	0.308	0.152	0.216	0.388	0.227	0.319	0.125	0.087	0.117	0.249	0.147	0.218	0.014	0.028	0.043
QF [69]	0.350	0.185	0.271	0.395	0.231	0.330	0.135	0.096	0.128	0.248	0.149	0.221	0.016	0.025	0.038
n(σ_x) [21]	0.250	0.178	0.258	0.027	0.017	0.036	0.035	0.020	0.027	0.212	0.138	0.194	0.175	0.147	0.217
RSD [56]	0.400	0.212	0.301	0.227	0.168	0.232	0.030	0.032	0.034	0.152	0.103	0.175	0.156	0.126	0.173
SMV [63]	0.154	0.099	0.137	0.074	0.058	0.062	0.217	0.112	0.149	0.210	0.189	0.265	0.008	0.028	0.047
NQC [60]	0.156	0.099	0.134	0.075	0.055	0.055	0.211	0.110	0.154	0.204	0.179	0.247	0.017	0.036	0.051
UEF^{NQC} [59]	0.216	0.121	0.18	0.002	0.023	0.029	0.054	0.009	0.027	0.115	0.042	0.067	0.176	0.133	0.202
NeuralQPP [66]	0.243	0.145	0.186	0.225	0.098	0.134	0.219	0.079	0.111	0.211	0.149	0.227	0.062	0.051	0.065
NQA-QPP [33]	0.072	0.039	0.069	0.075	0.08	0.116	0.064	0.058	0.076	0.105	0.055	0.078	0.038	0.006	0.002
BERT-QPP [5]	0.306	0.081	0.151	**0.414**	**0.248**	**0.374**	0.428	0.325	0.480	0.177	0.155	0.230	0.063	0.060	0.087
qpp-PRP [61]	0.107	0.081	0.085	0.208	0.177	0.271	0.042	0.025	0.035	0.103	0.077	0.113	0.235	0.207	**0.295**
NN-QPP [27]	0.372	0.163	0.260	0.338	0.191	0.274	0.441	0.342	0.450	0.170	0.150	0.226	0.019	0.039	0.080
CA-QPP	**0.411**	**0.230**	**0.327**	0.372	0.209	0.315	**0.448**	**0.347**	**0.482**	**0.310**	**0.237**	**0.336**	**0.259**	**0.218**	0.271

Table 4. Performance of CA-QPP using different backbone language models (BERT, DeBERTa, and MiniLM) across five datasets (TREC DL 2019, TREC DL 2020, DL-Hard, TREC DL 2021, and TREC DL 2022). Results are reported for all three correlation metrics (Pearson's ρ, Kendall's τ, and Spearman's ρ). The results show that DeBERTa achieves the highest performance on most datasets and metrics, while MiniLM, despite its smaller size, remains competitive

	TREC DL 2019			TREC DL 2020			DL-Hard			TREC DL 2021			TREC DL 2022		
Model	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$
bert	0.468	0.257	0.382	0.522	0.359	0.521	0.472	0.395	0.544	0.303	0.207	0.302	0.215	0.111	0.163
DeBERTa	0.583	0.377	0.541	0.531	0.334	0.471	0.545	0.452	0.5996	0.397	0.297	0.412	0.430	0.279	0.391
minilm	0.547	0.340	0.504	0.518	0.357	0.522	0.505	0.407	0.564	0.278	0.296	0.429	0.363	0. 210	0.297

language understanding. We also use MiniLM which is a lightweight model with significantly fewer parameters compared to BERT and DeBERTa, making it computationally efficient. Despite its size, it has shown competitive performance in ranking tasks due to its distilled architecture [65]. The experiments were conducted using the same setup and hyperparameters as described in Section 5.1 and Table 1.

The results for each model across the five datasets (TREC DL 2019, TREC DL 2020, DL-Hard, TREC DL 2021, and TREC DL 2022) are shown in Table 4. We observe that all three models demonstrate strong performance across the datasets and correlation metrics ($p-\rho$, $k-\tau$, $s-\rho$), indicating the robustness of CA-QPP. Notably, DeBERTa achieves the highest performance across most datasets and metrics, including TREC DL 2019, DL-Hard, and also maintains an advantage on the more recent TREC DL 2021 and TREC DL 2022. This can be attributed to its advanced architecture and improved contextual representations. On the other hand, MiniLM, despite its smaller size, shows competitive results, particularly on DL-Hard and TREC DL 2021, confirming that the effectiveness of CA-QPP does not rely on the scale of the language model to achieve strong performance.

In conclusion, in response to **RQ2**, we show that all three language models performed well when used in CA-QPP, showing robustness of our method w.r.t the backbone language model. Among the language models, DeBERTa emerged as the best-performing backbone language model, and we adopted it for the comparisons against other baselines. The results also highlight that CA-QPP is adaptable to smaller models like MiniLM, making it flexible for various use cases and computational environments.

5.4 Chameleon Queries

To rigorously assess the discriminative capacity of our proposed CA-QPP model for low performance queries, we evaluate its performance on the Chameleon query subset introduced by Arabzadeh et al. [8]. The Chameleon benchmark isolates a cohort of 6,980 MS MARCO queries that consistently rank among the lowest-performing across both sparse (e.g., BM25) and neural (e.g., DeepCT, ANCE, RepBERT, DocT5Query, TCT-ColBERT) retrievers. Within this set, the Lesser Chameleon subset comprises 1,693 queries that fall into the bottom 50% of performance across all six retrieval models, thereby offering a robust testbed for identifying systematic failure cases in modern IR pipelines.

To evaluate CA-QPP's sensitivity to query difficulty, we computed predicted performance scores for all 1,693 Lesser Chameleon queries and compared their distribution against predictions for the remaining 5,287 queries in the MS MARCO development set. Figure 4 visualizes the distributions of predicted scores across these two groups. The model assigns significantly lower scores to Lesser Chameleon queries, with a median predicted score of approximately 0.07 and an interquartile range (IQR) of roughly 0.05. In contrast, predictions for the broader development set are centered around a higher median of approximately 0.30, with a wider IQR close to 0.20. The distinct compression of scores in the Lesser Chameleon group indicates that CA-QPP consistently identifies these queries as uniformly difficult, and its narrow variance suggests high confidence in its predictions.

This result is especially important given that Chameleon queries have been shown to defy improvement across multiple model families and training methods. As noted by Arabzadeh et al. [8], these queries often reflect inherent ambiguity, lack of grounding in corpus content, or semantic sparsity, factors that are resistant to representation learning alone. The fact that CA-QPP assigns low confidence to such queries suggests that it encodes features predictive of retrieval hardness beyond surface-level term statistics. As such, CA-QPP not only achieves accurate average performance estimation across standard queries but also exhibits calibrated predictions on the hardest known subsets. Its ability to separate these two difficult and non-difficult query types shows its utility as a robust tool for predicting query performance.

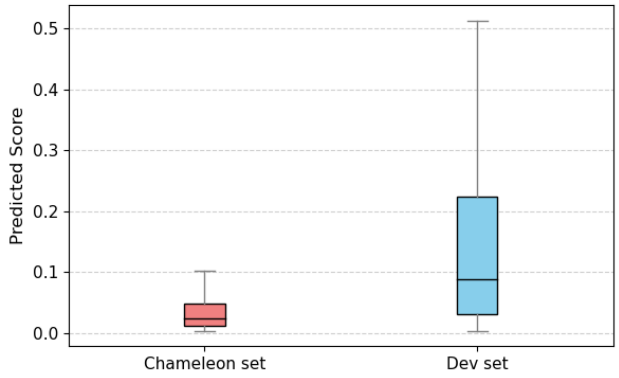


Fig. 4. The study of the performance of our model on Chameleon Queries

5.5 The Impact of Expansion Methods

In this section, we analyze the impact of different expansion methods on the performance of CA-QPP. Below, we summarize the configurations of the expansion methods evaluated:

- CA-QPP$_q$: Only the query terms are expanded, and the document terms are left unchanged.
- CA-QPP$_d$: Only the document terms are expanded, while the query terms remain unaffected.
- CA-QPP+: Expansion is performed only using positive weights.
- CA-QPP−: Expansion is performed only using negative weights.
- CA-QPP: This is our full model, where both the query and documents are expanded using both positive and negative weights.

Table 5. Comparison of the impact of expansion variations on the TREC DL 2019, TREC DL 2020, DL-Hard, TREC DL 2021 and TREC DL 2022 datasets.

	TREC DL 2019			TREC DL 2020			DL-Hard			TREC DL 2021			TREC DL 2022		
Method	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$
CA-QPP$_q$	0.600	0.398	0.559	0.507	0.323	0.462	0.514	0.413	0.582	0.265	0.235	0.350	0.387	0.190	0.256
CA-QPP$_d$	0.520	0.341	0.492	0.482	0.309	0.442	0.496	0.402	0.566	0.298	0.288	0.403	0.411	0.167	0.253
CA-QPP+	0.618	0.426	0.582	0.474	0.315	0.440	0.462	0.385	0.529	0.333	0.319	0.465	0.422	0.208	0.297
CA-QPP−	0.590	0.400	0.573	0.492	0.323	0.462	0.431	0.343	0.478	0.286	0.280	0.404	0.429	0.213	0.290
CA-QPP	0.583	0.377	0.541	0.531	0.334	0.471	0.545	0.452	0.600	0.379	0.297	0.412	0.430	0.279	0.391

As shown in Table 5, all expansion methods demonstrate reliable performance across the datasets. However, certain observations are worth highlighting:

1. **Positive vs. Negative Weights**: Expanding with only positive or only negative weights generally shows slightly lower performance than the full model, particularly on more challenging datasets such as DL-Hard. For instance, on DL-Hard, the full model achieves the highest values across all metrics ($p-\rho$ of 0.545, $k-\tau$ of 0.452, $s-\rho$ of 0.600), outperforming both CA-QPP+ and CA-QPP−. This pattern extends to TREC DL 2021 and TREC DL 2022, where using both types of weights together consistently results in stronger overall performance.

2. **Query vs. Document Expansion**: Between CA-QPP$_q$ and CA-QPP$_d$, we observe that expanding the query terms is more effective than expanding the document terms. We hypothesize that this is because documents might be noisier since they might not be exactly relevant to the query. On the other hand, queries although short, they are more accurate representation of the user's information need and less noisy. Therefore, although both variations work well, CA-QPP$_q$ is a slightly more accurate indicator of performance compared to CA-QPP$_d$.

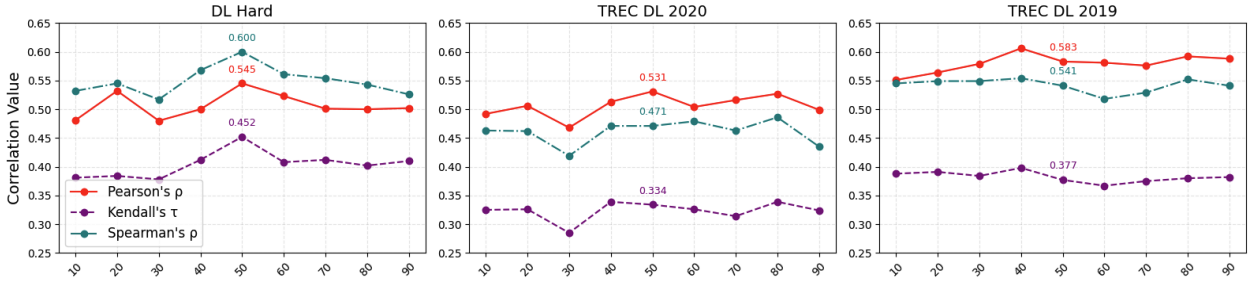
3. **Full Expansion CA-QPP**: The full model, which expands both query and document terms using both positive and negative weights, consistently performs well across all datasets and metrics. Although CA-QPP+may show slightly better performance in specific cases, such as on TREC DL 2019, our results indicate that CA-QPP provides the most robust and consistent performance across all datasets.

Additionally, we explore the impact of the weighting function parameter α on the performance of CA-QPP. As discussed in Section 3.4, α determines the scaling of term weights during the expansion process, effectively controlling the influence of expansion terms. To evaluate its effect, we varied α in the range $\{10, 20, \ldots, 100\}$ and measured the performance on all five datasets. The results, summarized in Figure 5, reveal that CA-QPP demonstrates robust performance across all datasets and metrics, even as α varies significantly. This robustness indicates that CA-QPP is not overly sensitive to the choice of α. While the performance remains stable across most values of α, tuning its value can further optimize results. We selected $\alpha = 50$ as the optimal value since it achieves a balanced performance across all datasets and metrics. For example, on DL-Hard and TREC DL 2020, $\alpha = 50$ maximizes performance across Pearson ($p-\rho$), Kendall ($k-\tau$), and Spearman ($s-\rho$) correlations. On TREC DL 2019, the performance for $\alpha = 50$ is comparable to the maximum observed performance ($\alpha = 40$). Similarly, for TREC DL 2021 and TREC DL 2022, $\alpha = 50$ yields performance that is close to the respective optima, making it a suitable and balanced choice across all datasets.

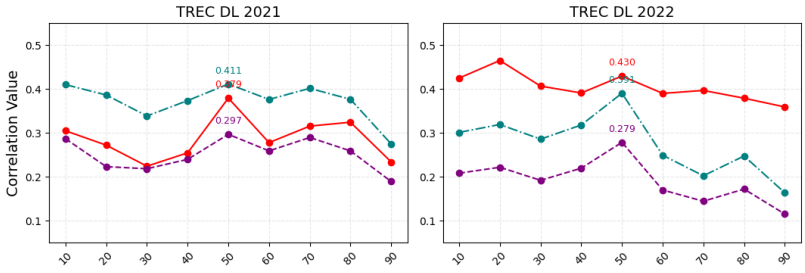
5.6 Impact of Aggregation Function

In this section, we investigate the impact of the aggregation function on CA-QPP's final predictions. As introduced earlier, the final performance prediction of CA-QPP is derived from an aggregation of two components: Query-Aware and Query-Agnostic. To understand how robust CA-QPP is to the choice of aggregation function, we evaluate its performance using different aggregation strategies, minimum, maximum, mean, and Reciprocal Rank Fusion (RRF) [10, 11, 14, 15].

Our hypothesis is that the minimum function provides a conservative (pessimistic) estimate, as it focuses on the weakest prediction between the two models. Conversely, the maximum function offers an optimistic estimate by predicting the highest potential performance among the two components. On the other hand, the mean function



(a) DL-Hard, TREC DL 2019, and TREC DL 2020 datasets.



(b) TREC DL 2021 and TREC DL 2022 datasets.

Fig. 5. Impact of expansion factor (α) x-axis on the performance of CA-QPP, measured by correlation metrics (Pearson's ρ, Kendall's τ, and Spearman's ρ) across various datasets.

provides a balanced, midpoint perspective. In addition, we leveraged RRF, which has been widely recognized in the literature for its effectiveness in ranking tasks, especially when fusing results from multiple models.

The results, presented in Figure 6, show that across most datasets and correlation metrics, the choice of aggregation function does not lead to significant differences in performance, indicating that CA-QPP is generally robust to this design choice. Notably, on DL-Hard with Pearson correlation, RRF achieves the highest scores among the aggregation methods. Moreover, RRF consistently delivers slightly better performance across datasets when evaluated using Pearson's ρ, whereas for Kendall's τ and Spearman's ρ, no single aggregation function emerges as consistently superior. These observations suggest that while all aggregation strategies yield comparable results in most cases, adopting RRF provides a modest advantage without introducing substantial trade-offs.

In response to **RQ4**, we conclude that while the choice of aggregation function does not have a significant impact on CA-QPP 's overall effectiveness, RRF demonstrates marginally better performance and is thus a reliable choice for combining the outputs of the `Query-Aware` and `Query-Agnostic` strategies.

6 Discussion

In this section, we provide an in-depth analysis of the performance of CA-QPP by focusing on its term weight prediction mechanism. Specifically, Section 6.1 examines the distribution of predicted term weights, comparing groups of queries that demonstrate high performance with those that perform poorly. This analysis highlights the predictive distinctions in term contributions between these query groups. In Section 6.2, we present illustrative examples of queries, their top-retrieved documents, and the associated predicted term weights. These examples showcase the effectiveness of CA-QPP in providing meaningful and interpretable predictions through actual values and outcomes.

6.1 Distribution of Estimated Weights

In this section, we analyze the predictive behavior of our models in assigning weights to query terms, focusing on differences between easy and hard queries. The analysis aims to understand how term weights—categorized as

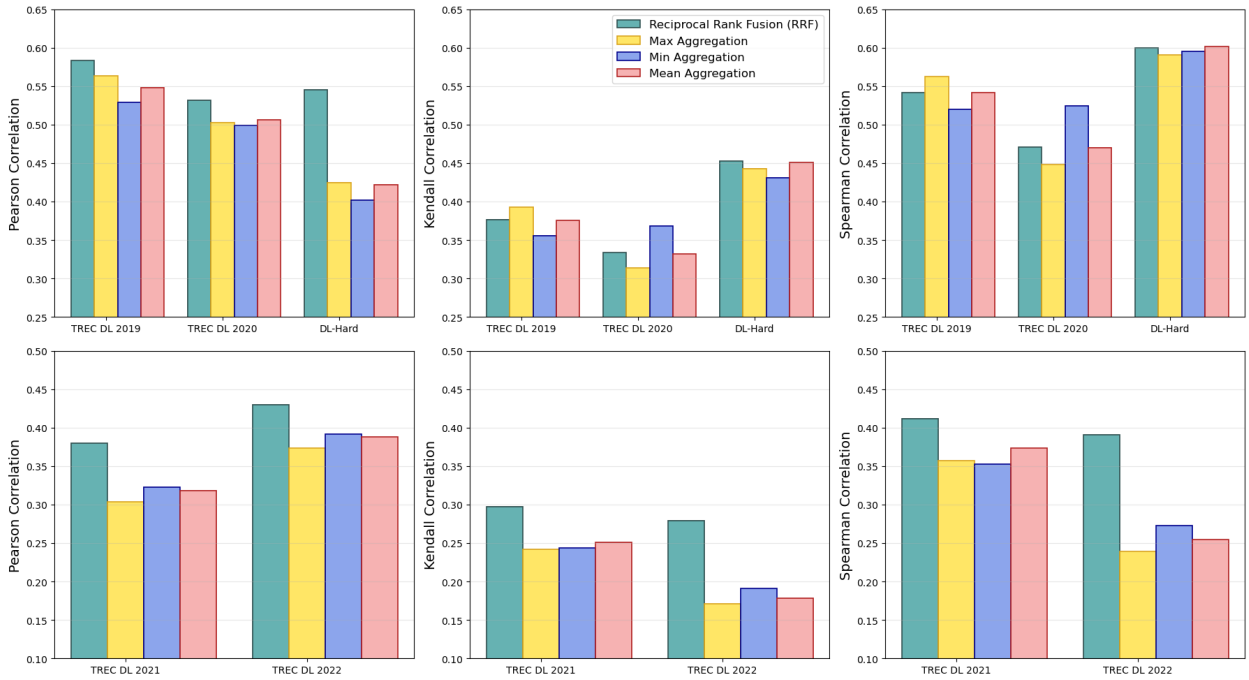


Fig. 6. Comparison of aggregation methods (Reciprocal Rank Fusion, Maximum Aggregation, Minimum Aggregation, and Mean Aggregation) on the performance of CA-QPP using Pearson, Kendall, and Spearman correlation metrics across the TREC DL 2019, TREC DL 2020, DL-Hard, TREC DL 2021 and TREC DL 2022 datasets.

promotive (positive weights) and *demotive* (negative weights)—vary across queries with significantly different retrieval outcomes. For this purpose, we categorize queries into two groups based on their Mean Average Precision (MAP) scores:

- *Easy Queries*: Defined as queries with nDCG scores ≥ 0.95, representing scenarios where the retrieval system performs exceptionally well.
- *Hard Queries*: Defined as queries with nDCG scores < 0.25, representing challenging scenarios with low retrieval effectiveness.

We randomly selected 5,000 queries from each category, ensuring no overlap with the training data, and analyzed the distributions of term weights assigned by the `Query-Aware` and `Query-Agnostic`. These results are visualized in Figure 7, which illustrates the distributions of positive and negative term weights for each query group. We present our observations as follows:

Distribution Trends for Easy Queries: (1) In both the `Query-Aware` (top row, left column) and `Query-Agnostic` (bottom row, left column), easy queries display a dominant frequency of positive term weights. This emphasizes that these queries primarily consist of *promotive terms* that enhance retrieval performance. (2) The mean absolute values of positive weights are significantly higher than those of negative weights, suggesting a strong alignment between query terms and relevant documents. The distribution of positive weights is concentrated toward higher values, reinforcing the promotive nature of the terms in easy queries.

Distribution Trends for Hard Queries: (1) For hard queries (right columns), there is a noticeable shift toward more negative term weights. In both models, the mean absolute values of negative weights exceed those of positive weights. This indicates that terms in hard queries tend to degrade retrieval performance, either by introducing ambiguity or misalignment with the underlying document collection. (2) The distributions show a higher presence of *demotive terms* in hard queries, highlighting the challenges these terms pose for effective document retrieval.

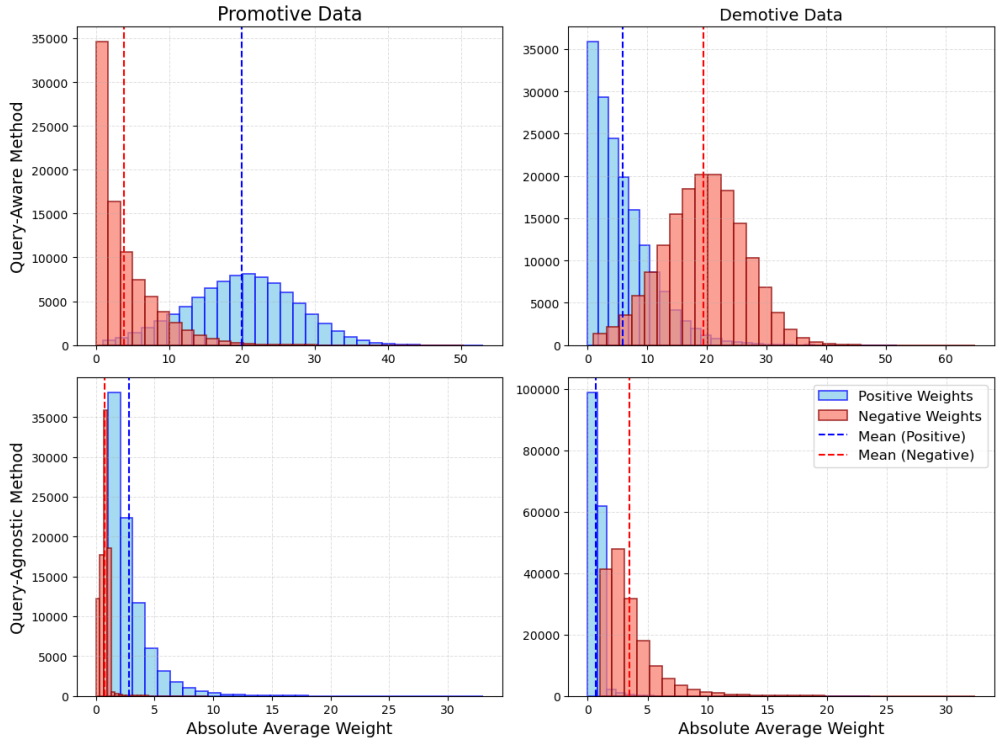


Fig. 7. Comparison of the distribution of the absolute average positive and negative weight of the Easy and Hard queries using the `Query-Aware` and `Query-Agnostic` strategies.

Comparison Between `Query-Aware` *and* `Query-Agnostic`*:* (1) The `Query-Aware` exhibits more balanced distributions, with narrower tails for both positive and negative weights. This suggests a more uniform evaluation of term contributions, likely because this model directly compares terms across retrieved documents and the relevance judgment. (2) The `Query-Agnostic`, in contrast, exhibits a longer-tailed distribution, particularly for negative weights. This suggests that the term importance scores have higher variance, reflecting a greater sensitivity to outlier terms or edge cases where certain terms drastically hinder retrieval. (3) The observed variability in the `Query-Agnostic` may stem from its reliance on retrieved documents without directly incorporating relevance judgments. As retrieved documents can vary significantly in relevance, structure, and content, the model assigns a broader range of weights to terms, capturing the diverse roles that terms play in influencing relevance.
Model Robustness: (1) The distinction between positive and negative term weights across easy and hard queries demonstrates that both models effectively capture query difficulty. For easy queries, the higher emphasis on *promotive terms* aligns with their role in retrieving relevant documents. For hard queries, the dominance of *demotive terms* reflects the challenges posed by poorly aligned or ambiguous terms. (2) The differences in distribution patterns between the `Query-Aware` and `Query-Agnostic` offer complementary insights. While the `Query-Aware` provides a stable, document-alignment-focused perspective, the `Query-Agnostic` captures a broader, more variable view of term contributions, potentially offering a richer signal for certain types of queries.

These findings validate the ability of both models to assign meaningful term weights that reflect the underlying retrieval effectiveness of queries. The visualized trends confirm the robustness of the term weighting process and the models' ability to differentiate between easy and hard queries based on term contributions.

Table 6. Sample queries and their retrieved documents highlighted based on predicted term weights. Darker blue represents promotive terms and darker red indicates demotive terms.

Query	Retrieved Document	Relevant?
behaviorism was the view that psychology should scientifically study behavior without	behaviorism was the view that psychology (1) should be an objective science (2) that studies behavior without reference to mental processes . onsciousness orignially started as observations of behavior , the view that psychology (1) should be an objective science that (2) studies behavior without reference to mental processes .	Yes
are dogs more intelligent than cats	overall , dogs are smarter than cats actually by quite a bit . however , cats have many more brain cells dedicated to sensory nerves (rather than used for storing information like dogs) . this means that even though dogs are much more intelligent than cats , cats are able to hear , smell , and see much better than dogs .	Yes
how long does it take to get a doctorate in psychology	thread: how long does it take to get urine pregnancy reults from the doctor ? how long does it take to get urine pregnancy reults from the doctor ? just as above really , I was just wondering how long does it take to get pregnancy results from a doctors urine test .	No

6.2 Case Study

In this section, we conduct a case study to evaluate the interpretability of term weights predicted by CA-QPP, particularly focusing on how terms are classified as promotive or demotive. To illustrate this, we present three sample queries from the MS MARCO dataset in Table 6, highlighting terms and their associated weights as predicted by CA-QPP. In the visualizations, terms are color-coded based on their weights: blue highlights promotive terms, and red indicates demotive terms. The intensity of the colors corresponds to the absolute magnitude of the weights, with darker shades representing higher weights. Importantly, both the term weights and colors are normalized for comparison. We describe our observations from this case study as follows:

(1) The purpose of the first query sample is to show that our proposed term weighting function is context-dependent and therefore, the weights that it generates for the same terms may differ across different queries. As seen in the table, the term *psychology* appears to be promotive in one query and demotive in another.
(2) The purpose of the query example "are dogs more intelligent than cats" is to illustrate how our method emphasizes terms that enhance retrieval effectiveness in high-performing queries. As seen in the table, terms such as *intelligent* are highlighted as promotive, aligning strongly with the query context. In contrast, terms like *dogs* and *cats* are less distinguishing, reflecting CA-QPP's focus on identifying contextually relevant information that contributes to effective retrieval.

(3) In the last example, its clear that the retrieval method has returned a completely irrelevant document, which is about 'pregnancy' rather than being about the length of a PhD program in Psychology. As such, the term weighting method has identified that pregnancy is an irrelevant term in this context and has assigned a highly demotive score to this term. On the other hand, it has emphasized the need to promote the term Psychology in the query itself.

These examples demonstrate how CA-QPP effectively assigns context-sensitive term weights that reflect the relevance and intent of the query. The contrast between promotive and demotive terms highlights the method's ability to adapt dynamically to different query-document contexts. This case study demonstrates how the method effectively differentiates between terms that enhance retrieval performance and those that hinder it, showcasing its ability to adapt to varying query contexts.

7 Concluding Remarks

In this paper, we presented CA-QPP, a novel approach for post-retrieval Query Performance Prediction (QPP) that leverages a contrastive framework to estimate query effectiveness. Building on the premise that the same information need can be expressed through multiple query formulations with varying levels of retrieval performance, we systematically construct two deliberate query variations: an *effective variation*, which emphasizes terms that enhance retrieval, and an *ineffective variation*, which amplifies terms that hinder retrieval. By contrasting the retrieval outcomes of these variations, CA-QPP provides a structured means of analyzing term-level contributions to query performance and predicting the effectiveness of the original query. Central to our approach is the use of a context-sensitive term-weighting mechanism to classify query terms as promotive, demotive, or neutral based on their influence on retrieval outcomes. These classifications form the basis for constructing the query variations, which are subsequently processed by a cross-encoder model to predict query performance. Our framework not only captures the nuanced interplay of terms within queries but also highlights the robustness and variability of retrieval systems in handling different query formulations.

We evaluated CA-QPP on the MS MARCO V1 and MS MARCO V2 datasets and their associated query sets, including TREC DL 2019, TREC DL 2020, DL-Hard, TREC DL 2021,and TREC DL 2022, which feature diverse query difficulties and extensive relevance judgments. Through comparisons with both traditional and neural QPP baselines, we demonstrated the effectiveness and stability of CA-QPP across standard correlation metrics. Additionally, our analysis of term weights and query-document alignments provides valuable interpretability, showing how individual terms shape retrieval effectiveness and contribute to query success or failure. The findings of this work emphasize the importance of understanding term-level contributions to retrieval effectiveness, particularly for queries that pose challenges to existing retrieval systems. By systematically identifying and contrasting effective and ineffective query formulations, CA-QPP offers a robust and interpretable framework for estimating query performance.

While CA-QPP demonstrates strong potential in post-retrieval QPP, there are several technical directions to further refine and expand its capabilities: (1) An interesting direction for future work involves integrating external behavioral data, such as clickthrough information, into the term-weighting process. Datasets like ORCAS, which include rich user interaction data, could be utilized to refine the identification of promotive and demotive terms by incorporating user preferences and relevance feedback. By aligning model predictions with actual user behavior, this approach could enhance the robustness of term-weight estimation, particularly for queries with ambiguous intent or sparse relevance judgments. Such integration would also provide insights into how user interactions shape retrieval performance, allowing for more informed predictions. (2) Another promising avenue is to improve the term-weighting process by explicitly capturing the interactions between query terms and their relationships with document terms. Instead of treating terms independently, advanced techniques like multi-headed attention mechanisms or graph neural networks (GNNs) could be used to model dependencies such as term co-occurrence, positional relationships, and semantic alignment. This would allow for a deeper understanding of how combinations of terms collectively influence retrieval effectiveness, enabling the construction of more precise query variations and enhancing performance prediction accuracy. Incorporating these relationships could lead to more sophisticated models capable of handling complex query formulations and challenging retrieval scenarios.

References

[1] Negar Arabzadeh, Amin Bigdeli, Radin Hamidi Rad, and Ebrahim Bagheri. 2023. Quantifying Ranker Coverage of Different Query Subspaces. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Taipei, Taiwan) *(SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 2298–2302. https://doi.org/10.1145/3539618.3592045

[2] Negar Arabzadeh, Amin Bigdeli, Shirin Seyedsalehi, Morteza Zihayat, and Ebrahim Bagheri. 2021. Matches Made in Heaven: Toolkit and Large-Scale Datasets for Supervised Query Reformulation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 4417–4425.

[3] Negar Arabzadeh, Radin Hamidi Rad, Maryam Khodabakhsh, and Ebrahim Bagheri. 2023. Noisy Perturbations for Estimating Query Difficulty in Dense Retrievers. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management* (Birmingham, United Kingdom) *(CIKM '23)*. Association for Computing Machinery, New York, NY, USA, 3722–3727. https://doi.org/10.1145/3583780.3615270

[4] Negar Arabzadeh, Radin Hamidi Rad, Maryam Khodabakhsh, and Ebrahim Bagheri. 2023. Noisy Perturbations for Estimating Query Difficulty in Dense Retrievers. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management* (Birmingham, United Kingdom) *(CIKM '23)*. Association for Computing Machinery, New York, NY, USA, 3722–3727. https://doi.org/10.1145/3583780.3615270

[5] 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 & Knowledge Management* (Virtual Event, Queensland, Australia) *(CIKM '21)*. Association for Computing Machinery, New York, NY, USA, 2857–2861. https://doi.org/10.1145/3459637.3482063

[6] Negar Arabzadeh, Chuan Meng, Mohammad Aliannejadi, and Ebrahim Bagheri. 2024. Query Performance Prediction: From Fundamentals tonbsp;Advanced Techniques. Springer-Verlag, Berlin, Heidelberg, 381–388. https://doi.org/10.1007/978-3-031-56069-9_51

[7] Negar Arabzadeh, Chuan Meng, Mohammad Aliannejadi, and Ebrahim Bagheri. 2024. Query Performance Prediction: Techniques and Applications in Modern Information Retrieval *(SIGIR-AP 2024)*. Association for Computing Machinery, New York, NY, USA, 291–294. https://doi.org/10.1145/3673791.3698438

[8] Negar Arabzadeh, Bhaskar Mitra, and Ebrahim Bagheri. 2021. MS MARCO Chameleons: Challenging the MS MARCO Leaderboard with Extremely Obstinate Queries. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 4426–4435.

[9] Negar Arabzadeh, Mahsa Seifikar, and Charles LA Clarke. 2022. Unsupervised Question Clarity Prediction Through Retrieved Item Coherency. In *CIKM*. 3811–3816.

[10] Negar Arabzadeh, Fattane Zarrinkalam, Jelena Jovanovic, Feras Al-Obeidat, and Ebrahim Bagheri. 2020. Neural embedding-based specificity metrics for pre-retrieval query performance prediction. *Information Processing & Management* 57, 4 (2020), 102248.

[11] Negar Arabzadeh, Fattaneh Zarrinkalam, Jelena Jovanovic, and Ebrahim Bagheri. 2019. Geometric estimation of specificity within embedding spaces. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2109–2112.

[12] Razieh Baradaran, Razieh Ghiasi, and Hossein Amirkhani. 2022. A survey on machine reading comprehension systems. *Natural Language Engineering* 28, 6 (2022), 683–732.

[13] David Carmel and Elad Yom-Tov. 2010. Estimating the query difficulty for information retrieval. In *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, Geneva, Switzerland, July 19-23, 2010*. 911. https://doi.org/10.1145/1835449.1835683

[14] David Carmel and Elad Yom-Tov. 2010. Estimating the query difficulty for information retrieval. *Synthesis Lectures on Information Concepts, Retrieval, and Services* 2, 1 (2010), 1–89.

[15] Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. 758–759.

[16] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, and Daniel Campos. 2021. Overview of the TREC 2020 deep learning track. *CoRR* abs/2102.07662 (2021). arXiv:2102.07662 https://arxiv.org/abs/2102.07662

[17] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Jimmy Lin. 2021. Overview of the TREC 2021 Deep Learning Track. In *Proceedings of the Thirtieth Text REtrieval Conference (TREC 2021)*. National Institute of Standards and Technology (NIST). https://trec.nist.gov/pubs/trec30/papers/Overview-DL.pdf

[18] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, Jimmy Lin, Ellen M. Voorhees, and Ian Soboroff. 2022. Overview of the TREC 2022 Deep Learning Track. In *Proceedings of the Thirty-First Text REtrieval Conference (TREC 2022)*. National Institute of Standards and Technology (NIST). https://trec.nist.gov/pubs/trec31/papers/Overview_deep.pdf

[19] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M Voorhees. 2020. Overview of the TREC 2019 deep learning track. *arXiv preprint arXiv:2003.07820* (2020).

[20] 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*. 299–306.

[21] 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*. 1089–1090.

[22] Zhuyun Dai and Jamie Callan. 2019. Context-aware sentence/passage term importance estimation for first stage retrieval. *arXiv preprint arXiv:1910.10687* (2019).

[23] Suchana Datta, Debasis Ganguly, Mandar Mitra, and Derek Greene. 2022. A Relative Information Gain-based Query Performance Prediction Framework with Generated Query Variants. *ACM Trans. Inf. Syst.* 41, 2, Article 38 (Dec. 2022), 31 pages. https://doi.org/10.1145/3545112

[24] Suchana Datta, Sean MacAvaney, Debasis Ganguly, and Derek Greene. 2022. A'Pointwise-Query, Listwise-Document'based Query Performance Prediction Approach. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

2148–2153.

[25] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[26] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805* (2019).

[27] Sajad Ebrahimi, Maryam Khodabakhsh, Negar Arabzadeh, and Ebrahim Bagheri. 2024. Estimating Query Performance Through Rich Contextualized Query Representations. In *European Conference on Information Retrieval*. Springer, 49–58.

[28] Guglielmo Faggioli, Nicola Ferro, Cristina Ioana Muntean, Raffaele Perego, and Nicola Tonellotto. 2023. A Geometric Framework for Query Performance Prediction in Conversational Search. In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval* (Taipei, Taiwan) *(SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 1355–1365. https://doi.org/10.1145/3539618.3591625

[29] Guglielmo Faggioli, Thibault Formal, Simon Lupart, Stefano Marchesin, Stephane Clinchant, Nicola Ferro, and Benjamin Piwowarski. 2023. Towards Query Performance Prediction for Neural Information Retrieval: Challenges and Opportunities. In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval* (Taipei, Taiwan) *(ICTIR '23)*. Association for Computing Machinery, New York, NY, USA, 51–63. https://doi.org/10.1145/3578337.3605142

[30] Guglielmo Faggioli, Thibault Formal, Stefano Marchesin, Stéphane Clinchant, Nicola Ferro, and Benjamin Piwowarski. 2023. Query Performance Prediction for Neural IR: Are We There Yet?. In *European Conference on Information Retrieval*. Springer, 232–248.

[31] Guglielmo Faggioli, Oleg Zendel, J. Shane Culpepper, Nicola Ferro, and Falk Scholer. 2022. sMARE: a new paradigm to evaluate and understand query performance prediction methods. *Inf. Retr.* 25, 2 (June 2022), 94–122. https://doi.org/10.1007/s10791-022-09407-w

[32] Luyu Gao and Jamie Callan. 2022. Unsupervised Corpus Aware Language Model Pre-training for Dense Passage Retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (Eds.). Association for Computational Linguistics, Dublin, Ireland, 2843–2853. https://doi.org/10.18653/v1/2022.acl-long.203

[33] Helia Hashemi, Hamed Zamani, and W Bruce Croft. 2019. Performance Prediction for Non-Factoid Question Answering. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval*. 55–58.

[34] 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 2008, Napa Valley, California, USA, October 26-30, 2008*. 1419–1420. https://doi.org/10.1145/1458082.1458311

[35] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. DeBERTa: Decoding-enhanced BERT with Disentangled Attention. *arXiv preprint arXiv:2006.03654* (2020).

[36] Seyed Mohammad Hosseini, Negar Arabzadeh, Morteza Zihayat, and Ebrahim Bagheri. 2024. Enhanced Retrieval Effectiveness through Selective Query Generation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. 3792–3796.

[37] Parastoo Jafarzadeh and Faezeh Ensan. 2022. A semantic approach to post-retrieval query performance prediction. *Information Processing & Management* 59, 1 (2022), 102746.

[38] K Sparck Jones, Steve Walker, and Stephen E. Robertson. 2000. A probabilistic model of information retrieval: development and comparative experiments: Part 2. *Information processing & management* 36, 6 (2000), 809–840.

[39] Samreen Kazi, Shakeel Khoja, and Ali Daud. 2023. A survey of deep learning techniques for machine reading comprehension. *Artificial Intelligence Review* 56, Suppl 2 (2023), 2509–2569.

[40] Oren Kurland, Anna Shtok, David Carmel, and Shay Hummel. 2011. A unified framework for post-retrieval query-performance prediction. In *Advances in Information Retrieval Theory: Third International Conference, ICTIR 2011, Bertinoro, Italy, September 12-14, 2011. Proceedings 3*. Springer, 15–26.

[41] Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. 2020. Contrastive representation learning: A framework and review. *Ieee Access* 8 (2020), 193907–193934.

[42] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2356–2362.

[43] Jimmy Lin, Rodrigo Nogueira, and Andrew Yates. 2021. Pretrained transformers for text ranking: Bert and beyond. *Synthesis Lectures on Human Language Technologies* 14, 4 (2021), 1–325.

[44] Iain Mackie, Jeffrey Dalton, and Andrew Yates. 2021. How Deep is your Learning: the DL-HARD Annotated Deep Learning Dataset. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[45] Chuan Meng. 2024. Query Performance Prediction for Conversational Search and Beyond. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Washington DC, USA) *(SIGIR '24)*. Association for Computing Machinery, New York, NY, USA, 3077. https://doi.org/10.1145/3626772.3657658

[46] Chuan Meng, Negar Arabzadeh, Mohammad Aliannejadi, and Maarten de Rijke. 2023. Query Performance Prediction: From Ad-hoc to Conversational Search. *arXiv preprint arXiv:2305.10923* (2023).

[47] Chuan Meng, Negar Arabzadeh, Arian Askari, Mohammad Aliannejadi, and Maarten de Rijke. 2024. Query Performance Prediction using Relevance Judgments Generated by Large Language Models. arXiv:2404.01012 [cs.IR] https://arxiv.org/abs/2404.01012

[48] Chuan Meng, Guglielmo Faggioli, Mohammad Aliannejadi, Nicola Ferro, and Josiane Mothe. 2025. QPP++ 2025: Query Performance Prediction andnbsp;Its Applications innbsp;thenbsp;Era ofnbsp;Large Language Models. In *Advances in Information Retrieval: 47th European Conference on Information Retrieval, ECIR 2025, Lucca, Italy, April 6–10, 2025, Proceedings, Part V* (Lucca, Italy). Springer-Verlag, Berlin, Heidelberg,

319–325. https://doi.org/10.1007/978-3-031-88720-8_49

[49] Ricardo Marçal de Andrade Nascimento, Daniel Xavier de Sousa, Guglielmo Faggioli, Paulo José Lage Alvarenga, Nicola Ferro, and Marcos André Gonçalves. 2025. A Robustness Assessment of Query Performance Prediction (QPP) Methods Based on Risk-Sensitive Analysis. In *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR)* (Padua, Italy) *(ICTIR '25)*. Association for Computing Machinery, New York, NY, USA, 416–429. https://doi.org/10.1145/3731120.3744611

[50] 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. In *CoCo@ NIPS*.

[51] Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. 2019. From doc2query to docTTTTTquery. *Online preprint* 6 (2019), 2.

[52] Ronak Pradeep, Rodrigo Nogueira, and Jimmy Lin. 2021. The expando-mono-duo design pattern for text ranking with pretrained sequence-to-sequence models. *arXiv preprint arXiv:2101.05667* (2021).

[53] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research* 21, 140 (2020), 1–67. http://jmlr.org/papers/v21/20-074.html

[54] Fiana Raiber and Oren Kurland. 2014. Query-performance prediction: setting the expectations straight. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. 13–22.

[55] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *CoRR* abs/1908.10084 (2019). arXiv:1908.10084 http://arxiv.org/abs/1908.10084

[56] 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, ICTIR 2017, Amsterdam, The Netherlands, October 1-4, 2017*, Jaap Kamps, Evangelos Kanoulas, Maarten de Rijke, Hui Fang, and Emine Yilmaz (Eds.). ACM, 245–248. https://doi.org/10.1145/3121050.3121087

[57] Sara Salamat, Negar Arabzadeh, Shirin Seyedsalehi, Amin Bigdeli, Morteza Zihayat, and Ebrahim Bagheri. 2023. Neural Disentanglement of Query Difficulty and Semantics. In *CIKM*. 4264–4268.

[58] Abbas Saleminezhad, Negar Arabzadeh, Reza Hosseini Rad, and Ebrahim Bagheri. 2025. Robust Query Performance Prediction for Dense Retrievers via Adaptive Disturbance Generation. *Machine Learning* 114, 65 (2025). https://doi.org/10.1007/s10994-024-06659-z

[59] 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*. 259–266.

[60] Anna Shtok, Oren Kurland, and David Carmel. 2016. Query Performance Prediction Using Reference Lists. *ACM Trans. Inf. Syst.* 34, 4 (2016), 19:1–19:34. https://doi.org/10.1145/2926790

[61] Ashutosh Singh, Debasis Ganguly, Suchana Datta, and Craig McDonald. 2023. Unsupervised Query Performance Prediction for Neural Models with Pairwise Rank Preferences. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Taipei, Taiwan) *(SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 2486–2490. https://doi.org/10.1145/3539618.3592082

[62] Atsushi Sugiura and Oren Etzioni. 2000. Query routing for web search engines: Architecture and experiments. *Computer Networks* 33, 1-6 (2000), 417–429.

[63] 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*. 1891–1894.

[64] Nicola Tonellotto, Craig Macdonald, and Iadh Ounis. 2013. Efficient and effective retrieval using selective pruning. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining* (Rome, Italy) *(WSDM '13)*. Association for Computing Machinery, New York, NY, USA, 63–72. https://doi.org/10.1145/2433396.2433407

[65] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems* 33 (2020), 5776–5788.

[66] 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 & Development in Information Retrieval*. 105–114.

[67] Oleg Zendel, J Shane Culpepper, and Falk Scholer. 2021. Is query performance prediction with multiple query variations harder than topic performance prediction?. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1713–1717.

[68] Oleg Zendel, Anna Shtok, Fiana Raiber, Oren Kurland, and J. Shane Culpepper. 2019. Information Needs, Queries, and Query Performance Prediction *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 395–404. https://doi.org/10.1145/3331184.3331253

[69] Yun Zhou and W Bruce Croft. 2007. Query performance prediction in web search environments. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. 543–550.