BERT-QPP: Contextualized Pre-trained Transformers for Query Performance Prediction

Negar Arabzadeh
University of Waterloo
narabzad@uwaterloo.ca

Maryam Khodabakhsh
Shahrood University of Technology
m_khodabakhsh@shahroodut.ac.ir

Ebrahim Bagheri
Ryerson University
bagheri@ryerson.ca

ABSTRACT

Query Performance Prediction (QPP) is focused on estimating the difficulty of satisfying a user query for a certain retrieval method. While most state of the art QPP methods are based on term frequency and corpus statistics, more recent work in this area have started to explore the utility of pretrained neural embeddings, neural architectures and contextual embeddings. Such approaches extract features from pretrained or contextual embeddings for the sake of training a supervised performance predictor. In this paper, we adopt contextual embeddings to perform performance prediction, but distinguish ourselves from the state of the art by proposing to directly fine-tune a contextual embedding, i.e., BERT, specifically for the task of query performance prediction. As such, our work allows the fine-tuned contextual representations to estimate the performance of a query based on the association between the representation of the query and the retrieved documents. We compare the performance of our approach with the state-of-the-art based on the MS MARCO passage retrieval corpus and its three associated query sets: (1) MS MARCO development set, (2) TREC DL 2019, and (3) TREC DL 2020. We show that our approach not only shows significant improved prediction performance compared to all the state-of-the-art methods, but also, unlike past neural predictors, it shows significantly lower latency, making it possible to use in practice.

CCS CONCEPTS

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

KEYWORDS

Query Performance Prediction, Contextualized Pre-trained transformers, Cross-encoder, Bi-encoder

ACM Reference Format:
Negar Arabzadeh, Maryam Khodabakhsh, and Ebrahim Bagheri . 2021. BERT-QPP: Contextualized Pre-trained Transformers for Query Performance Prediction. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, QLD, Australia.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3459637.3482063

CIKM '21, November 1–5, 2021, Virtual Event, QLD, Australia

ACM ISBN 978-1-4503-8446-9/21/11…$15.00
https://doi.org/10.1145/3459637.3482063

1 INTRODUCTION

The ad hoc retrieval task is focused on retrieving a ranked list of documents from a large document corpus to satisfy an information need expressed through a query. Despite the strides made on the ad hoc retrieval tasks ranging from more traditional term statistics and query likelihood language models [16, 20] to more recent deep neural rankers [18, 40], there are always some group of difficult queries that cannot be effectively satisfied by a retrieval method. For this reason, the objective of the Query Performance Prediction (QPP) task is to predict the possible performance of a retrieval method on a given input user query. In other words, the QPP task aims to predict whether the retrieval method will be able to retrieve relevant documents for a certain query without having access to gold standard relevance information for that query. An effective QPP method can play an important role in the ad hoc retrieval process such as enabling query routing or query reformulation [5, 34].More recent applications of QPP methods can be seen in the area of intelligent assistants, in which users are quite sensitive to the response they receive from the assistant[31].

Query performance prediction methods have often been broadly categorized as either pre-retrieval or post-retrieval methods [13]. Pre-retrieval methods focus on predicting the performance of a query solely based on the association between the user query and the information available in the document corpus. In contrast, post-retrieval methods consider the query, the document corpus, as well as the initial set of documents that are retrieved for the query from the document corpus by the retrieval method [5]. Existing post-retrieval methods leverage a host of potential signals that can point to the difficulty of a query such as the association between the query and the retrieved documents, the relation between the query and the corpus, or that of the corpus and the retrieved documents, as well as considering the distribution of scores associated with the retrieved documents [30]. Empirically speaking, post-retrieval methods that rely on the retrieval score of the retrieved documents have shown promising results. We note that one of the deficiencies of these methods is their reliance on term statistics within both the query and document spaces [1, 8, 9, 26, 35–37, 43]; hence, they do not necessarily perform as effectively when vocabulary mismatch exists between the two spaces. In order to address the vocabulary mismatch problem, researchers have adopted different forms of pre-trained neural representations to perform both ad hoc retrieval as well as query performance prediction [2, 2–4, 15, 19, 23, 33, 41]. Such approaches show significantly better retrieval effectiveness and performance prediction accuracy compared to their non-neural counterparts. Despite notable performance, methods that rely on pre-trained neural embeddings are not able to distinguish between potentially differing semantics associated with the same surface form of a term as they are context-independent. For this reason,

more recent approaches, such as NQA-QPP [12] employ contextualized pre-trained embeddings such as BERT [10] to address the QPP task, overcoming the drawbacks of pre-trained neural embeddings and hence showing improved performance. Specifically, NQA-QPP consists of three key components, namely retrieval score component, query component, and query-document component, which map retrieved documents scores as well as contextualized BERT representations of queries and documents into a d-dimensional space. Further, all of the three components are aggregated and fed into a feed-forward neural network to address QPP.

In this paper, we advance the state of the art by adopting contextualized neural embeddings for query performance prediction. Our work proposes to fine-tune a pre-trained transformer to learn to estimate the performance of a query in light of the set of documents that have been retrieved for that query. We propose an architecture, referred to as BERT-QPP, that directly learns query performance through the fine-tuning of BERT. As opposed to other BERT-based models used in ad hoc retrieval [17, 40, 42], our proposed BERT-QPP approach is focused on learning the quality of retrieved documents, and hence as a result, learns the potential performance of the query, based on the retrieval score of the retrieved documents. This allows BERT-QPP to learn query performances directly through the fine-tuning of BERT without the need for additional training. We consider two widely used architectures, namely cross-encoder and bi-encoder architectures, to operationalize BERT-QPP. We fine-tune BERT in both architectures using queries and their associated retrieved documents. BERT-QPP distinguishes itself from the state of the art NQA-QPP method [12] in that in the context of NQA-QPP, BERT representations of queries and documents are considered to be input features of a second feed-forward neural network, which is trained to predict the performance of the queries. In contrast, by fine-tuning BERT in our proposed framework, BERT-QPP can directly learn to predict query performance based on the retrieved documents and as such, our proposed approach does not require a separate trained model in addition to a fine-tuned BERT. This makes our proposed approach computationally less demanding and lighter weight to deploy. In addition, most QPP methods suffer from high sensitivity w.r.t their hyperparameter. However, our proposed BERT-QPP method does not have any parameters, and consequently it is more robust. To show the utility of our proposed QPP approach, we conduct experiments on the well-known MS MARCO passage retrieval dataset [24] in addition to TREC Deep Learning Track 2019 and 2020 queries, which are judged under different judging schemes to show that having incomplete labels does not affect the training of our BERT-QPP approach. We show that our proposed approach is able to show improved performance prediction performance across all query sets. based on Pearson ρ, Kendall τ and Spearman ρ Correlation metrics.

2 PROPOSED APPROACH

Let us first formally define the QPP task in the context of ad hoc retrieval. Given a collection of documents C, a list of retrieved documents D_q and an input query q, and a retrieval method R, a query performance predictor μ needs to estimate the performance of R on q with respect to a desirable metric M, e.g., average precision or reciprocal rank. The predictor μ can be defined as follows:

$$\widehat{M} \leftarrow \mu(q, D_q, C) \tag{1}$$

In order to realize μ, we propose to fine-tune a contextualized pre-trained transformer, i.e., BERT using a regression model that would output a continuous score exhibiting the difficulty of the query for the retrieval methodwithin two architectures, namely a cross-encoder network and a bi-encoder network.

2.1 Cross-Encoder Network

For training a cross-encoder network for developing an efficient μ, we learn a continuous difficulty score based on the association between the input query and the top-k documents retrieved by R in response to q. To do so, we concatenate the input query and the top-k retrieved documents, i.e., D_q^k, using the special separator token. We apply a linear layer on the first vector produced by the transformer, in order to produce a scalar value referred to as $QPP_{Cross}(q, D_q^k)$. We leverage a sigmoid layer and a one-class Binary Cross Entropy loss function. More formally, the loss function for our cross-encoder network can be defined as follows where $M(q, D_q)$ is the desired ranking metric such as average precision:

$$\begin{aligned} l(QPP_{cross}(q, D_q^K), M(q, D_q)) = \\ -w[M(q, D_q).\ log\ \sigma(QPP_{cross}(q, D_q^K)) + \\ (1-M(q, D_q).\ log\ (1 - \sigma(QPP_{cross}(q, D_q^K)))] \end{aligned} \tag{2}$$

The advantage of having all the query and the top-k documents in a single transformer is that there will be higher interactions between the query and the top-k documents, which can potentially capture deeper associations between the query and document spaces. However, this comes at the expense of an increased computational cost [14, 29, 38, 39]. In addition, the model requires the computation of the association between the query and the top-k documents at runtime and as soon as the query is received. This prevents any possible offline computation in advance as those are contextually derived contingent upon the input query. Thus it can face slow inference time in practice.

2.2 Bi-Encoder Network

To address the possible shortcomings of the cross-encoder network, it is possible to adopt a bi-encoder network architecture where the query q and the set of top-k retrieved documents D_q^k are fed into a Siamese network architecture which consists of two parallel towers, namely T_q and T_D where T_q is the tower that learns a representation of query q and T_D is the tower that will learn document representations. As such, $T_q(q)$ and $T_D(D_q^k)$ will denote the representations for the query and the top-k retrieved documents, respectively. Given such representations, it is possible to learn the association between the query and the top-k retrieved documents through some similarity measure. Here without the loss of generality, we utilize cosine similarity to minimize the loss $l(q, D_q^k)$ as follows where $Sim_{cos}(X, Y)$ represents the cosine similarity between vectors X and Y:

$$\begin{aligned} QPP_{bi}(q, D_q^K) = Sim_{cos}(T_q(q), T_D(D_q^K)) \\ l(QPP_{bi}(q,D_q^K), M(q, D_q)) = ||M(q, D_q) - QPP_{bi}(q, D_q^K)||_2 \end{aligned} \tag{3}$$

Compared to the cross-encoder architecture, the bi-encoder drastically reduces the computation overhead. This is primarily because the two towers of the architecture start with the same pre-trained

network; however, they will be updated separately during the fine-tuning process and after fine-tuning neither the representations of the query nor the top-k retrieved documents are not dependent on each other. Considering the number of interactions learnt between the query and top-k retrieved documents are fewer compared to the cross-encoder architecture, it can be expected that the performance of the bi-encoder would be lower than the cross-encoder network.

We note the major novelty of our work is that while cross-encoder and bi-encoder architectures have been used in the past [11, 14, 25, 29] for learning the relevance of documents to an input query based on labeled training data, our work does not learn relevance between the query and the judged relevant documents but rather learns the success of the retrieved top-k documents in addressing the query. As such, both loss functions in Equations 2 and 3 consist of $M(q, D_q)$, which is a measure of performance of R on q. Thus, our trained architectures are able to estimate how well the query has been satisfied by R as opposed to being able to rank documents and produce suitable representations for QPP task.

3 EXPERIMENTS

3.1 Dataset and Evaluation

We empirically study the performance of our proposed approach on the well-known MS MARCO passage collection [24] which consists of 8.8 million passages and is accompanied by more than 500k query and relevant document pairs. In addition, there are 6,980 queries that are intended to be used for evaluation purposes, which is known as the MS MARCO Development Set (dev set). In addition to the MS MARCO dev set queries, we consider queries from the TREC Deep Learning Track 2019 and 2020 [6, 7]. The former includes 43 queries and the latter consists of 53 queries. The main difference between the two deep learning track query sets and the MS MARCO dev set is their scale and relevance judgement schema. While the number of queries in MS MARCO dev set is quite large , they were sparsely labeled, i.e., one relevant judged document per query on average. However, within the two deep learning sets, each query has been judged thoroughly on a 4-level scale. We selected these three sets of queries to study the robustness of BERT-QPP w.r.t different query set sizes and relevance judgement schema.

For evaluation purposes, as suggested in the literature [5, 13], we predict the performance of queries when retrieving 1,000 documents per query by a well-known first stage retriever BM25 [16]. To measure performance, we use the official metric of the MS MARCO leaderboard, i.e., MRR@10. Due to space limitations, we resort to reporting the MAP@10 results in our Github repository[1]. The common approach for evaluating the performance of a QPP method is to use correlation metrics between the ranked set of queries based on their predicted difficulty compared to their actual difficulty [5]. We measure linear correlation by Pearson's ρ and ranking correlation by Kendall's τ and Spearman ρ coefficient. Higher correlations reflect more accurate query performance prediction.

3.2 Baselines

We compare our proposed method against several post-retrieval query performance predictors that have already shown promising results in the literature [5]. WIG [43] predicts query difficulty by measuring the divergence between retrieval score of top documents and the collection of documents. Clarity [8] operates by measuring the divergence between the language model of the retrieved documents and the corpus. Query Feedback (QF) [43] measures the overlap between the top retrieved documents from the original query and a revised query using low-expense and straightforward techniques such as pseudo-relevance feedback [21]. The majority of post-retrieval QPP metrics, which have shown the highest performance across most of the benchmarks [5], employ the standard deviation of the retrieval score of top retrieved documents. NQC [36],σ_k [27], n($\sigma_{\%}$) [9] and SMV [37] and RSD [32] benefit from a variation of such standard deviation. The Utility Estimation Framework (UEF) [35] operates over well-performing QPP baseline methods such as NQC [36]. Furthermore, we consider three state-of-the-art supervised QPP baselines. NeuralQPP [41] is a framework which uses other unsupervised QPP methods as signals for weakly-supervised learning. Similarly, LTRoq [28] is a supervised QPP method that utilizes a learning to rank framework and considers other QPP methods as features. We also include the latest BERT-based QPP method, namely NQA-QPP [12] and note that it is the latest contextualized pre-trained embedding based QPP approach in the literature. Since most of the introduced QPP baselines require hyper-parameter tuning, we randomly held out a small subset of the queries from the MS MARCO training set (5,000 queries) for tuning the baseline hyper-parameters. Further, we tuned the baseline hyperparameters for TREC 2019 on TREC 2020 queries and vice versa.

3.3 Experimental Setup

We used Google's BERT Base pre-trained transformer model with 12 layers and attention heads with 768 dimensions of final representations to implement BERT-QPP. We trained each architecture for one epoch on the queries of the MSMARCO training set with a batch size of 16. Training on a higher number of epochs did not show any further improvements. Without loss of generality, we feed the query and first retrieved document to the transformer due to 1) transformer input length limitations, and 2) since we want to be able to precompute the document embeddings in the bi-encoder architecture. We believe that the first retrieved document indicates the quality of the retrieved list especially in collections such as MS MARCO, whose relevance judgements are sparse.

3.4 Results and Findings

We report our findings of the experiments in Table 1 which shows the correlation values over three query sets. We make important observations based on these results: (1) we find that our approach shows the highest correlations based on the MS Marco dev set. This query set consists of 6,980 queries. This observation can also be made on the NQA-QPP method, which is also based on contextual embeddings as well. This can indicate that models such as ours that are based on fine-tuning of contextual embedding show improved performance over large query sets. (2) we note that the cross-encoder variation of our proposed approach shows consistently better performance over all of the baselines and the three query sets. This is especially important as our proposed approach does not require the tuning of any hyper-parameters and shows such performance out-of-the-box while all other baselines have been fine-tuned for their best performance. (3) we observe that unlike the baselines, both

[1] https://github.com/Narabzad/BERTQPP

Table 1: Performance of our proposed QPP methods vs Baselines on MSMARCO in terms of Pearson ρ Kendall τ and Spearman ρ correlation. All the correlations mentioned in this table are statistically significant with MRR@10 ($\alpha = 0.05$, two-tailed paired t-test). Bold values indicate outperforming the best of the baselines.

	MS MARCO (6,980 queries)			TREC DL 2019 (43 queries)			TREC DL 2020 (53 queries)		
Method	Pearson ρ	Kendall τ	Spearman ρ	Pearson ρ	Kendall τ	Spearman ρ	Pearson ρ	Kendall τ	Spearman ρ
Clarity	0.149	0.258	0.345	0.137	0.145	0.193	0.235	0.264	0.338
WIG	0.154	0.170	0.227	0.186	0.115	0.133	0.255	0.221	0.289
QF	0.170	0.210	0.264	0.172	0.148	0.183	0.233	0.211	0.241
LTRoq	0.171	0.029	0.039	0.171	0.062	0.073	0.135	0.088	0.115
NeuralQPP	0.193	0.171	0.227	0.127	0.058	0.068	0.236	0.215	0.273
n($\sigma_{\%}$)	0.221	0.217	0.284	0.166	0.172	0.214	0.259	0.223	0.280
σ_k	0.250	0.256	0.339	0.135	0.148	0.189	0.306	0.263	0.327
RSD	0.310	0.337	0.447	0.235	0.188	0.247	0.250	0.301	0.379
SMV	0.311	0.271	0.357	0.219	0.215	0.274	0.266	0.221	0.280
NQC	0.315	0.272	0.358	0.226	0.165	0.215	0.261	0.263	0.333
UEF_{NQC}	0.316	0.303	0.398	0.268	0.225	0.284	0.271	0.282	0.338
NQA-QPP	0.451	0.364	0.475	0.155	0.202	0.260	0.321	0.283	0.356
BERT-QPP_{bi}	**0.473**	0.355	0.464	**0.339**	**0.338**	**0.421**	0.280	0.202	0.256
BERT-QPP_{cross}	**0.517**	**0.400**	**0.520**	**0.358**	**0.355**	**0.451**	**0.351**	**0.328**	**0.418**

Table 2: Comparing Neual-based QPP methods based on their runtime per query in milliseconds.

Method	Inference time per query (ms)
NQA-QPP	25.3
NeuralQPP	21.3
BERT-QPP_{cross}	2.6
BERT-QPP_{bi}	**0.68**

variations of our proposed approach show a stable performance on all three query sets and their performance does not show inferior performance compared to other baseline depending on the query set. However, this is not the case on the stronger baseline methods. For instance, while NQA-QPP shows strong performance on MS MARCO dev set and TREC DL 2020 query set, it does not show a competitive performance on TREC DL 2019. A similar observation can be made for other strong baseline, namely UEF_{NQC}, which shows strong performance on TREC DL 2019 but not as competitive on the other two query sets, and (4) as expected our proposed variation with the cross-encoder architecture shows more favorable results compared to the bi-encoder on MS MARCO dev set and TREC DL 2020 and better yet competitive results with our bi-encoder network variation on TREC DL 2019. As such, we believe this model would be preferred in terms of query performance prediction accuracy.

While we find the cross-encoder architecture outperforms bi-encoder network in general, we note that QPP methods need to not only be effective but also fast in practice. As such, we compare the inference time of the variations of our proposed approach with each other as well as the neural-based QPP baselines in Table 2 when run on an RTX3090 GPU. We put the inference times reported in Table 2 in context by mentioning that Lin et al. [22] have reported a 55ms per query latency on the BM25 retrieval method. We report several observations based on the inference times reported in this table: (1) When comparing the two variations of our approach, the bi-encoder architecture shows significantly lower inference time (4 times smaller) compared to the cross-encoder network. This can be primarily attributed to the fact that while the cross-encoder network needs to encode query and document representations in tandem and at runtime, the bi-encoder only requires the computation of the query representation as well as a cosine similarity operation computing the similarity of the query representation and the offline computed representation of the first retrieved document. As such, it will have much faster inference time. (2) compared to the query latency for retrieving relevant documents based on BM25 (55ms per query), the delay caused by our query performance prediction methods can be considered to be tolerable depending on the application domain. The lengthier variation of our approach, i.e., the cross-encoder introduces a 4.7% overhead while the bi-encoder introduces a 1.3% overhead to the retrieval process. For time-critical application domains, the bi-encoder network would be the preferred variation as it maintains competitive effectiveness while exhibiting fast inference time. For mission-critical applications, the cross-encoder architecture would be more desirable as it shows better overall performance on all of our query sets especially on MS MARCO dev set which consists of over six thousand queries. (3) We note that compared to the other two neural baseline methods, our slower variation, i.e., BERT-QPP_{cross}, is at least 8 times faster than the fastest neural baseline, i.e., NeuralQPP. This is an important observation as the baseline neural methods may not be practical to use in production given their high latency.

4 CONCLUSION

We have proposed a method to predict the performance of queries in the context of ad hoc retrieval based on the fine-tuning of a contextualized pre-trained transformer such that it learns to estimate the performance of a retrieved list of documents for a given input query. We proposed two variations namely BERT-QPP_{cross} and BERT-QPP_{bi} with cross-encoder and bi-encoder architectures, respectively. While the former enjoys a consistently strong performance over the baselines, the latter is more efficient with relatively small latency. Our experiments show that BERT-QPPs outperform state-of-the-art QPP baselines on three query sets associated with the MS MARCO collection. In addition, we show that both BERT-QPP_{cross} and BERT-QPP_{bi} are substantially faster than existing neural QPP methods.

REFERENCES

[1] Negar Arabzadeh, Amin Bigdeli, Morteza Zihayat, and Ebrahim Bagheri. 2021. Query Performance Prediction Through Retrieval Coherency. In *Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 - April 1, 2021, Proceedings, Part II (Lecture Notes in Computer Science)*, Djoerd Hiemstra, Marie-Francine Moens, Josiane Mothe, Raffaele Perego, Martin Potthast, and Fabrizio Sebastiani (Eds.), Vol. 12657. Springer, 193–200. https://doi.org/10.1007/978-3-030-72240-1_15

[2] 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.

[3] 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.

[4] Negar Arabzadeh, Fattane Zarrinkalam, Jelena Jovanovic, and Ebrahim Bagheri. 2020. Neural Embedding-Based Metrics for Pre-retrieval Query Performance Prediction. In *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14-17, 2020, Proceedings, Part II (Lecture Notes in Computer Science)*, Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins (Eds.), Vol. 12036. Springer, 78–85. https://doi.org/10.1007/978-3-030-45442-5_10

[5] 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.

[6] 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

[7] 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).

[8] 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.

[9] 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.

[10] 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).

[11] Yingqi Qu Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2020. RocketQA: An Optimized Training Approach to Dense Passage Retrieval for Open-Domain Question Answering. *arXiv preprint arXiv:2010.08191* (2020).

[12] 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.

[13] Claudia Hauff. 2010. Predicting the effectiveness of queries and retrieval systems. In *SIGIR Forum*, Vol. 44. 88.

[14] Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2019. Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring. *arXiv preprint arXiv:1905.01969* (2019).

[15] Ayyoob Imani, Amir Vakili, Ali Montazer, and Azadeh Shakery. 2019. Deep neural networks for query expansion using word embeddings. In *European Conference on Information Retrieval*. Springer, 203–210.

[16] 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.

[17] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906* (2020).

[18] Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 39–48.

[19] Maryam Khodabakhsh and Ebrahim Bagheri. 2021. Semantics-enabled query performance prediction for ad hoc table retrieval. *Information Processing & Management* 58, 1 (2021), 102399.

[20] John Lafferty and Chengxiang Zhai. 2001. Document language models, query models, and risk minimization for information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. 111–119.

[21] Victor Lavrenko and W. Bruce Croft. 2001. Relevance-Based Language Models. In *SIGIR 2001: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, September 9-13, 2001, New Orleans, Louisiana, USA*, W. Bruce Croft, David J. Harper, Donald H. Kraft, and Justin Zobel (Eds.). ACM, 120–127. https://doi.org/10.1145/383952.383972

[22] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2020. Distilling Dense Representations for Ranking using Tightly-Coupled Teachers. *arXiv preprint arXiv:2010.11386* (2020).

[23] Bhaskar Mitra, Nick Craswell, et al. 2018. *An introduction to neural information retrieval*. Now Foundations and Trends.

[24] 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*.

[25] Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. *arXiv preprint arXiv:1901.04085* (2019).

[26] Joaquín Pérez-Iglesias and Lourdes Araujo. 2010. Standard deviation as a query hardness estimator. In *International Symposium on String Processing and Information Retrieval*. Springer, 207–212.

[27] Joaquín Pérez-Iglesias and Lourdes Araujo. 2010. Standard Deviation as a Query Hardness Estimator. In *String Processing and Information Retrieval - 17th International Symposium, SPIRE 2010, Los Cabos, Mexico, October 11-13, 2010. Proceedings (Lecture Notes in Computer Science)*, Edgar Chávez and Stefano Lonardi (Eds.), Vol. 6393. Springer, 207–212. https://doi.org/10.1007/978-3-642-16321-0_21

[28] 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.

[29] Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084* (2019).

[30] Haggai Roitman. 2020. ICTIR Tutorial: Modern Query Performance Prediction: Theory and Practice. In *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval*. 195–196.

[31] Haggai Roitman, Shai Erera, and Guy Feigenblat. 2019. A Study of Query Performance Prediction for Answer Quality Determination. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval*. 43–46.

[32] 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

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

[34] Surendra Sarnikar, Zhu Zhang, and J Leon Zhao. 2014. Query-performance prediction for effective query routing in domain-specific repositories. *Journal of the Association for Information Science and Technology* 65, 8 (2014), 1597–1614.

[35] 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.

[36] Anna Shtok, Oren Kurland, David Carmel, Fiana Raiber, and Gad Markovits. 2012. Predicting query performance by query-drift estimation. *ACM Transactions on Information Systems (TOIS)* 30, 2 (2012), 1–35.

[37] 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.

[38] Jesse Vig and Kalai Ramea. 2019. Comparison of transfer-learning approaches for response selection in multi-turn conversations. In *Workshop on DSTC7*.

[39] Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019. Transfertransfo: A transfer learning approach for neural network based conversational agents. *arXiv preprint arXiv:1901.08149* (2019).

[40] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *arXiv preprint arXiv:2007.00808* (2020).

[41] 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.

[42] Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2020. RepBERT: Contextualized Text Embeddings for First-Stage Retrieval. *arXiv preprint arXiv:2006.15498* (2020).

[43] 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.