





Estimating Query Performance Through Rich Contextualized Query Representations

Sajad Ebrahimi¹, Maryam Khodabakhsh², Negar Arabzadeh³, and Ebrahim Bagheri⁴

¹ University of Guelph, Guelph, ON, Canada sebrah05@uoguelph.ca

² Shahrood University of Technology, Shahrood, Iran

m_khodabakhsh@shahroodut.ac.ir

³ University of Waterloo, Waterloo, ON, Canada narabzad@uwaterloo.ca

⁴ Toronto Metropolitan University, Toronto, ON, Canada bagheri@torontomu.ca

Abstract. The state-of-the-art query performance prediction methods rely on the fine-tuning of contextual language models to estimate retrieval effectiveness on a per-query basis. Our work in this paper builds on this strong foundation and proposes to learn rich query representations by learning the interactions between the query and two important contextual information, namely (1) the set of documents retrieved by that query, and (2) the set of similar historical queries with known retrieval effectiveness. We propose that such contextualized query representations can be more accurate estimators of query performance as they embed the performance of past similar queries and the semantics of the documents retrieved by the query. We perform extensive experiments on the MSMARCO collection and its accompanying query sets including MSMARCO Dev set and TREC Deep Learning tracks of 2019, 2020, 2021, and DL-Hard. Our experiments reveal that our proposed method shows robust and effective performance compared to state-of-the-art baselines.

1 Introduction

Information Retrieval (IR) researchers have been concerned with both the *effectiveness* and *robustness* of retrieval methods [14,6,33]. A successful IR method would be one that simultaneously shows both effective and robust performance, i.e., shows strong and consistent performance over a large range of queries. While not ideal, but in practice, IR methods are often only effective on a subset of queries and less effective on others. By identifying challenging queries for an IR method, it would be possible to adopt alternative strategies to satisfy these queries such as query routing [39], query reformulation [36], and asking users to clarify their intents [7,1]. The task of *Query Performance Prediction (QPP)* speaks directly to this need and focuses on estimating the effectiveness of retrieval methods on input queries.

Broadly speaking, QPP methods have been classified into the categories of *pre-retrieval* and *post-retrieval* methods [11]. The former is concerned with predicting the performance of a query prior to retrieval, whereas the latter counterparts consider additional information accessible after an initial retrieval stage,

such as retrieval scores of retrieved documents [41,44,9,38,8,10,2,3], among others. Given access to a wider range of information, post-retrieval methods often exhibit stronger performance and can themselves be seen as unsupervised [46,41,44,17,4,30] and supervised [45,20,5,18,25] variations. Most recently and thanks to large-scale relevance judgment collections such as MS MARCO [32], researchers have more extensively explored supervised post-retrieval QPP methods [45,20,5,18]. For example, NeuralQPP [45] was among the first neural frameworks that used unsupervised QPP methods as weak signals to learn a more effective supervised method. There has also been interest in employing contextualized Large Language Models (LLM) [12] in the QPP task where the performance of a query is estimated through the finetuning of an LLM [5,18,26]. Most existing supervised post-retrieval QPP methods focus on estimating the performance of a query by finetuning an LLM for this purpose [5,18]. The underlying assumption of these methods is that the semantic finetuned representation of the query obtained from an LLM may be correlated with the performance of the query.

Our work in this paper aligns closely with earlier works [5,18] and provides a more generalizable framework to learn rich and contextualized query representations that can more effectively estimate the performance of the query. We propose that a rich query representation suitable for query performance prediction would be one that is informed by the interaction between the query and (1) the documents retrieved by that query, and (2) the set of similar historical queries with known effectiveness. The underlying premise of our work is that contextual language models capture meaningful geometric relations [23,31,19]; therefore, contents that are placed closer to each other in the embedding space carry similar semantics and hence would exhibit similar characteristics, such as comparable retrieval effectiveness, when used in applications like retrieval. On this basis, learning rich contextualized representations for queries that are influenced by *relevant content from the document space* as well as *relevant content from the query space* can provide insight into the potential effectiveness of the query. Our work is guided by the hypothesis that queries that are embedded in close proximity to effective historical queries and semantically relevant documents are more likely to be effective queries themselves. In contrast, queries that are embedded in close proximity to ineffective queries and whose set of retrieved documents do not have a semantic resemblance to the query are more likely to be ineffective.

For this reason, we propose to learn rich contextualized query representations based on the interaction between the query, its retrieved documents, and past similar historical queries in order to predict the query’s potential effectiveness. In order to learn such rich representations, we propose to finetune a contextual language model to capture these interactions through a cross-encoder architecture. To show the effectiveness of our approach, we have performed extensive experiments on five widely used MS MARCO datasets, namely the Dev, TREC DL 2019, 2020, 2021 and DL-Hard sets [32,15,28,13]. Our experiments show that our method enjoys significantly higher effectiveness on the QPP task compared

to other state-of-the-art approaches. For reproducibility purposes, we made our code and model publicly available at GitHub ¹.

2 Proposed Approach

Problem Definition. Let C, q, R, D_q , be the collection of documents, input query, a retrieval method, and a ranked list of documents retrieved by R in response to query q , respectively. The task of QPP is concerned with developing a predictor μ to estimate the performance of R on q based on a given retrieval metric M , e.g., average precision or reciprocal rank, without accessing the relevance judgments. This can be expressed as: $\widehat{M}_q = \mu(q, D_q, C)$ where \widehat{M}_q is an estimated value of M for query q .

Hypothesis. The underlying premise of our work is that a rich contextualized representation for a query, which can effectively predict the performance of the query should not only consider the representation of the query itself but also capture (1) *The association between the retrieved documents by the query and the query*: Earlier research has shown that the qualities of the list of documents retrieved for a query, such as coherence [16], can be indicators of the possible effectiveness of that query. As such, a rich query representation that can encode and embody the characteristics of the retrieved set of documents for that query is more likely to effectively estimate query performance; and (2) *The relation with past similar queries with known effectiveness*: A rich query representation that can effectively identify past similar queries with comparable retrieval effectiveness will have a higher likelihood of estimating the effectiveness of the query based on its association with queries with analogous performance.

Proposed Formulation. To encode the above three characteristics in our rich query representation, we contextualize the query as follows: (1) We capture the individual characteristics of the query through its contextualized representation using a pre-trained language model. This ensures that queries that are both semantically and syntactically similar to each other receive comparable representations; (2) The properties of the retrieved documents are also considered by representing them through their contextualized representations, and (3) The association between each query and its most similar historical queries is also computed through their geometric distance in the embedding space. Two queries would be considered to be more similar if they have smaller distances from each other. Similar queries can be identified through a *nearest neighbor* scheme. We systematically incorporate this contextual information into our query representation through a cross-encoder architecture that finetunes a language model to estimate query performance. In particular, for query q , we let the cross-encoder architecture estimate the performance of q , through regression, based on the contextualized representation of q, D_q , and a set of most similar queries to q with known effectiveness \hat{Q}_q . We encode q, D_q , and \hat{Q}_q and their retrieval effectiveness

¹ <https://github.com/sadjadeb/nearest-neighbour-qpp>

Table 1. The performance of our proposed approach and baselines on Dev set with MRR@10 and DL-Hard with NDCG@10. The correlations are statistically significant with 95% confidence interval.

	MS MARCO Dev			DL-Hard		
	$p - \rho$	$k - \tau$	$s - \rho$	$p - \rho$	$k - \tau$	$s - \rho$
Clarity	0.149	0.258	0.345	0.149	0.099	0.126
WIG	0.154	0.170	0.227	0.331	0.260	0.348
QF	0.170	0.210	0.264	0.210	0.164	0.217
NeuralQPP	0.193	0.171	0.227	0.173	0.111	0.134
n($\sigma\%$)	0.221	0.217	0.284	0.195	0.120	0.147
RSD	0.310	0.337	0.447	0.362	0.322	0.469
SMV	0.311	0.271	0.357	0.375	0.269	0.408
NQC	0.315	0.272	0.358	0.384	0.288	0.417
UEF _{NQC}	0.316	0.303	0.398	0.359	0.319	0.463
NQA-QPP	0.451	0.364	0.475	0.386	0.297	0.418
BERT-QPP	0.517	0.400	0.520	0.404	0.345	0.472
qpp-BERT-PL	0.520	0.413	0.522	0.330	0.266	0.390
qpp-PRP	0.302	0.311	0.412	0.090	0.061	0.063
Ours	0.555	0.421	0.544	0.434	0.412	0.508

by concatenating them using a special separator token and then apply a linear layer on the first vector produced to estimate a scalar value of \widehat{M} as the difficulty of the query. We leverage a sigmoid layer and a one-class Binary cross-entropy loss function. Given $M(q, D_q)$ as the desired ranking metric, such as average precision, we adopt the following loss function to train for query performance:

$$\ell(\widehat{M}_q, M(q, D_q)) = -w[M(q, D_q).log(\sigma(\widehat{M}_q)) + (1 - M(q, D_q).log(1 - \sigma(\widehat{M}_q)))] \quad (1)$$

Nearest Neighbor Queries. In addition to the query itself, and the set of retrieved documents retrieved by R for q , our approach also requires access to Q_q . In order to identify the set of most similar queries to q , we first collect a set of historical queries with known effectiveness, namely $\chi = \{(q_1, M_1), (q_2, M_2), \dots, (q_n, M_n)\}$. Given a representation function f that maps each q_i to a vector in the embedding space, we define a query store QS consisting of a set of key-value pairs where each key is the embedding representation of a previously seen query and a corresponding value that denotes the performance of that query. The query store QS can be formulated as:

$$QS \stackrel{\text{def}}{=} \{(\kappa, v)\} = \{(f(q_i), M(q_i))\} \forall i \in \{1, \dots, n\} \quad (2)$$

The query store QS can be indexed using an approximate nearest neighborhood indexing mechanism [24], allowing for the efficient retrieval of \hat{Q}_q for query q . Given a distance function $\Psi(\cdot, \cdot)$, we first generate a contextualized representation of query $f(q)$ and then find the nearest neighbors of the query from QS .

Table 2. The performance of our proposed approach and baselines on Trec Deep Learning Track 2019, 2020, and 2021. The correlations are statistically significant on NDCG@10 with 95% confidence interval.

	2019			2020			2021		
	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$
Clarity	0.271	0.229	0.332	0.360	0.215	0.296	0.111	0.070	0.094
WIG	0.310	0.158	0.226	0.204	0.117	0.166	0.197	0.195	0.270
QF	0.295	0.240	0.340	0.358	0.266	0.366	0.132	0.101	0.142
NeuralQPP	0.289	0.159	0.224	0.248	0.129	0.179	0.134	0.221	0.188
$n(\sigma\%)$	0.371	0.256	0.377	0.480	0.329	0.478	0.269	0.169	0.256
RSD	0.460	0.262	0.394	0.426	0.364	0.508	0.256	0.224	0.340
SMV	0.495	0.289	0.440	0.450	0.391	0.539	0.252	0.192	0.278
NQC	0.466	0.267	0.399	0.464	0.294	0.423	0.271	0.201	0.292
UEF _{NQC}	0.507	0.293	0.432	0.511	0.347	0.476	0.272	0.223	0.327
NQA-QPP	0.348	0.164	0.255	0.507	0.347	0.496	0.258	0.185	0.265
BERT-QPP	0.491	0.289	0.412	0.467	0.364	0.448	0.262	0.237	0.34
qpp-BERT-PL	0.432	0.258	0.361	0.427	0.280	0.392	0.247	0.172	0.292
qpp-PRP	0.321	0.181	0.229	0.189	0.157	0.229	0.027	0.004	0.015
Ours	0.519	0.318	0.459	0.462	0.318	0.448	0.322	0.266	0.359

3 Experiments

Training Set. For our experiments, we adopt the well-known MS MARCO passage retrieval dataset [32], which consists of 8.8M passages. The training set includes over 500k search queries that correspond with at least one relevance-judged passage. We utilize this set of queries to build the query store QS and also to train our model. For each query q in this dataset, we obtain the retrieved documents D_q , as well as nearest neighbor queries Q_q , which are then used for training the cross-encoder architecture and estimating M_q .

Test Set. We evaluate our model on five query sets: (1) The MS MARCO Development set, also referred to as the Dev set, which consists of 6,980 queries. (2) The TREC Deep Learning Track 2019, [15], (3) The TREC Deep Learning Track 2020 [12], (4) The TREC Deep Learning Track 2021 [43], and (5) The TREC Deep Learning Hard set (DL-Hard) [29]. These query sets consist of 43, 54, 47, and 50 queries, respectively, and differ from the MS MARCO Dev set in that they provide multiple relevant judged passages for each query whereas the Dev set consists of, on average, one judged passage per query.

Evaluation Metrics. For evaluation purposes, we follow the well-known strategy of computing the correlation between the set of queries that are ranked based on their predicted performance against their actual performance based on the standard performance metric for each dataset, i.e., MRR@10 on the Dev set and NDCG@10 on the others. To this end, we compute Pearson ρ for linear correlation, Kendall τ , and Spearman ρ for rank correlation. A higher correlation value shows more accurate query performance prediction. For the retrieval method, we estimate BM25 implemented by Pyserini [27].

Experimental Setup. We use the pre-trained language model, DeBERTa [22], which has shown huge success in different downstream IR and Natural Lan-

guage processing tasks [21], to create vector representations and conduct nearest-neighbor sampling using Faiss [24]. In order to implement $\Psi(\cdot, \cdot)$, we adapt the widely-used cosine similarity distance. In order to train our model, we adopt the implementation offered for the cross-encoder architecture in the SentenceTransformers package [35]. Without loss of generality and as suggested in [5], we set $|D_q| = 1$ as well as $|\hat{Q}_q| = 1$ and we used Map@20 as for retrieval effectiveness labels of our model as well as the effectiveness of \hat{Q}_q . The model was trained on a 24GB NVIDIA GeForce RTX 3090 GPU with a batch size of 16 for one epoch, and the training process took an hour and a half.

Baselines. We compare our model against the state-of-the-art post-retrieval QPP baselines. These include the following methods: **Clarity** which works based on the KL divergence between language models induced from retrieved documents and the corpus. **WIG** [46], **NQC**[41], $n(\sigma\%)$ [17], **RSD** [37] and **SMV**[44] which are all score-based methods that predict query performance by computing different statistics of the retrieval scores of the top-ranked documents. Unlike the above predictors that are unsupervised, **NeuralQPP** [45] is the first supervised QPP method that uses existing unsupervised QPP methods as signals to perform weakly-supervised learning. The Utility Estimation Framework (UEF)[40] is designed to function alongside highly effective QPP baseline methods such as **NQC**. **NQA-QPP** [20], is another supervised method that uses a BERT model to learn the representations of queries and documents. **BERT-QPP** [5] also which fine-tunes BERT to directly predict the retrieval score of the query. **QppBERT-PL** [18] is one of the most recent BERT-based methods that uses point-wise training on individual queries, and list-wise training over top-ranked pseudo-relevant documents. Additionally, we consider **QPP-PRP** [42], which was designed to evaluate the performance of neural rankers by assessing the level of agreement between a pairwise neural reranker, exemplified by **DuoT5** [34], and the ranked list generated by the neural ranker for a given query.

Findings. The results of our experiments compared to the state-of-the-art baselines are shown in Tables 1 and 2. We can highlight the following findings based on the results in these two tables:

(1) Our proposed method has shown better performance compared to all baseline models on four of the five test sets. This includes the MS MARCO Dev set, which consists of the largest number of queries, and also the DL-Hard set, which includes challenging queries.

(2) On the DL 2020 dataset where our proposed approach does not show the best performance, we find that there is no single baseline that shows the best performance on all three metrics. In fact **NQA-QPP** shows the best performance on Pearson correlation while **SMV** shows better performance on Kendall and Spearman correlations. Therefore, there are no robust baselines for this dataset in the state of the art.

(3) We note that on the DL 2020 dataset where we do not outperform the baselines, our proposed approach has a similar performance to supervised baseline methods that fine-tune a contextual embedding model, i.e., **BERT-QPP**, **qpp-BERT-PL**, and **qpp-PRP**. In fact on this dataset, unsupervised baselines that

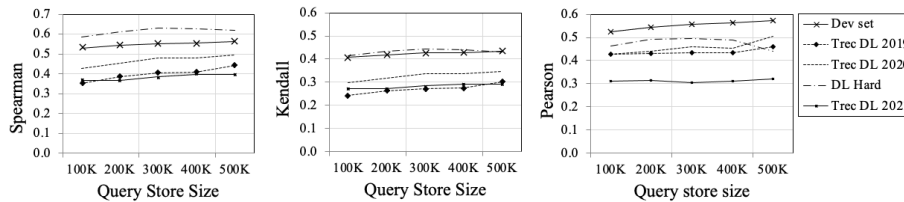


Fig. 1. The impact of the query store size on the performance of our approach.

do not use contextual embeddings such as SMV and RSD show better performance on rank correlation metrics, i.e., Spearman and Kendall correlations.

(4) In terms of robustness, our proposed approach shows the most consistent performance compared to all the baselines. For instance, SMV that shows a high rank correlation on DL 2020, does not show competitive performance on DL 2021 or MS MARCO Dev. Similarly, qpp-BERT-PL, which offers the best performance among the baselines on MS MARCO Dev, is not competitive on DL-Hard. However, our approach shows a consistent behavior across all datasets and metrics.

Finally, we explore the impact of the query store size on the performance of our proposed approach. Given we integrate most similar queries from the query store with known retrieval effectiveness, the size of the query store can have an impact on the performance of our model. We empirically study to what extent having a smaller query store size could have a negative impact on the performance of our method. To assess this, we randomly down-sampled the query store by only including 100k, 200k, 300k, 400k, and 500k from the MS MARCO training set. We report the impact of this down-sampling on all five query sets and using three correlation measures in Figure 1. As seen in the figure, while larger query stores lead to improved overall performance, smaller query stores still show competitive performance. In fact, we find that the differences between the smallest query store size and the largest are not statistically significant.

4 Concluding Remarks

In this paper, we have shown that a rich contextualized query representation that encodes the semantics of the query itself, as well as the interactions of the query with its set of retrieved documents, along with its most similar historical queries, can be quite effective for predicting the performance of the query. It means the model can perform better when it knows the performance of similar data to the given input. Our experiments performed on five widely used datasets show that our proposed approach offers strong and robust performance on a range of QPP metrics.

References

1. Aliannejadi, M., Kiseleva, J., Chuklin, A., Dalton, J., Burtsev, M.: Building and evaluating open-domain dialogue corpora with clarifying questions. In: EMNLP (2021)
2. Arabzadeh, N., Bigdeli, A., Zihayat, M., Bagheri, E.: 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 43. pp. 193–200. Springer (2021)
3. Arabzadeh, N., Bigdeli, A., Zihayat, M., Bagheri, E.: Query performance prediction through retrieval coherency. In: Hiemstra, D., Moens, M., Mothe, J., Perego, R., Potthast, M., Sebastiani, F. (eds.) 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, vol. 12657, pp. 193–200. Springer (2021). https://doi.org/10.1007/978-3-030-72240-1_15, https://doi.org/10.1007/978-3-030-72240-1_15
4. Arabzadeh, N., Hamidi Rad, R., Khodabakhsh, M., Bagheri, E.: Noisy perturbations for estimating query difficulty in dense retrievers. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. pp. 3722–3727 (2023)
5. Arabzadeh, N., Khodabakhsh, M., Bagheri, E.: Bert-qpp: contextualized pre-trained transformers for query performance prediction. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management. pp. 2857–2861 (2021)
6. Arabzadeh, N., Mitra, B., Bagheri, E.: Ms marco chameleons: challenging the ms marco leaderboard with extremely obstinate queries. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management. pp. 4426–4435 (2021)
7. Arabzadeh, N., Seifkar, M., Clarke, C.L.: Unsupervised question clarity prediction through retrieved item coherency. In: Proceedings of the 31st ACM International Conference on Information & Knowledge Management. pp. 3811–3816 (2022)
8. Arabzadeh, N., Zarrinkalam, F., Jovanovic, J., Al-Obeidat, F., Bagheri, E.: Neural embedding-based specificity metrics for pre-retrieval query performance prediction. *Information Processing & Management* **57**(4), 102248 (2020)
9. Arabzadeh, N., Zarrinkalam, F., Jovanovic, J., Bagheri, E.: 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 42. pp. 78–85. Springer (2020)
10. Arabzadeh, N., Zarrinkalam, F., Jovanovic, J., Bagheri, E.: Geometric estimation of specificity within embedding spaces. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management. pp. 2109–2112 (2019)
11. Carmel, D., Yom-Tov, E.: Estimating the query difficulty for information retrieval. *Synthesis Lectures on Information Concepts, Retrieval, and Services* **2**(1), 1–89 (2010)
12. Craswell, N., Mitra, B., Yilmaz, E., Campos, D.: Overview of the TREC 2020 deep learning track. *CoRR* **abs/2102.07662** (2021), <https://arxiv.org/abs/2102.07662>
13. Craswell, N., Mitra, B., Yilmaz, E., Campos, D.: Overview of the TREC 2020 deep learning track. *CoRR* **abs/2102.07662** (2021), <https://arxiv.org/abs/2102.07662>
14. Craswell, N., Mitra, B., Yilmaz, E., Campos, D., Lin, J.: Ms marco: Benchmarking ranking models in the large-data regime. In: Proceedings of the 44th International

- ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1566–1576 (2021)
15. Craswell, N., Mitra, B., Yilmaz, E., Campos, D., Voorhees, E.M.: Overview of the trec 2019 deep learning track. In: Text REtrieval Conference (TREC) (2020)
 16. Cronen-Townsend, S., Zhou, Y., Croft, W.B.: Predicting query performance. In: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. pp. 299–306 (2002)
 17. Cummins, R., Jose, J., O’Riordan, C.: Improved query performance prediction using standard deviation. In: Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. pp. 1089–1090 (2011)
 18. Datta, S., MacAvaney, S., Ganguly, D., Greene, D.: A ‘pointwise-query, listwise-document’ based qpp approach. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (2022). <https://doi.org/10.1145/3477495.3531821>
 19. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers). pp. 4171–4186. Association for Computational Linguistics (2019). <https://doi.org/10.18653/v1/n19-1423>, <https://doi.org/10.18653/v1/n19-1423>
 20. Hashemi, H., Zamani, H., Croft, W.B.: Performance prediction for non-factoid question answering. In: Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval. pp. 55–58 (2019)
 21. He, P., Gao, J., Chen, W.: Deberv3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. arXiv preprint arXiv:2111.09543 (2021)
 22. He, P., Liu, X., Gao, J., Chen, W.: Deberv: Decoding-enhanced bert with disentangled attention. In: International Conference on Learning Representations (2021), <https://openreview.net/forum?id=XPZiaotutsD>
 23. Hofmann, V., Pierrehumbert, J.B., Schütze, H.: Dynamic contextualized word embeddings. arXiv preprint arXiv:2010.12684 (2020)
 24. Johnson, J., Douze, M., Jégou, H.: Billion-scale similarity search with GPUs. IEEE Transactions on Big Data **7**(3), 535–547 (2019)
 25. Khodabakhsh, M., Bagheri, E.: Semantics-enabled query performance prediction for ad hoc table retrieval. Inf. Process. Manag. **58**(1), 102399 (2021). <https://doi.org/10.1016/J.IPM.2020.102399>, <https://doi.org/10.1016/j.ipm.2020.102399>
 26. Khodabakhsh, M., Bagheri, E.: Learning to rank and predict: Multi-task learning for ad hoc retrieval and query performance prediction. Information Sciences **639**, 119015 (2023)
 27. Lin, J., Ma, X., Lin, S.C., Yang, J.H., Pradeep, R., Nogueira, R.: 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. pp. 2356–2362 (2021)
 28. Mackie, I., Dalton, J., Yates, A.: 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 (2021)

29. Mackie, I., Dalton, J., Yates, A.: 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. pp. 2335–2341 (2021)
30. Meng, C., Arabzadeh, N., Aliannejadi, M., de Rijke, M.: Query performance prediction: From ad-hoc to conversational search. arXiv preprint arXiv:2305.10923 (2023)
31. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
32. Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., Deng, L.: Ms marco: A human generated machine reading comprehension dataset. In: CoCo@ NIPs (2016)
33. Penha, G., Câmara, A., Hauff, C.: Evaluating the robustness of retrieval pipelines with query variation generators. In: European conference on information retrieval. pp. 397–412. Springer (2022)
34. Pradeep, R., Nogueira, R., Lin, J.: The expando-mono-duo design pattern for text ranking with pretrained sequence-to-sequence models (2021)
35. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics (2019)
36. Roitman, H., Erera, S., Feigenblat, G.: A study of query performance prediction for answer quality determination. In: Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval. pp. 43–46 (2019)
37. Roitman, H., Erera, S., Weiner, B.: Robust standard deviation estimation for query performance prediction. In: Kamps, J., Kanoulas, E., de Rijke, M., Fang, H., Yilmaz, E. (eds.) Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR 2017, Amsterdam, The Netherlands, October 1-4, 2017. pp. 245–248. ACM (2017). <https://doi.org/10.1145/3121050.3121087>, <https://doi.org/10.1145/3121050.3121087>
38. Salamat, S., Arabzadeh, N., Seyedsalehi, S., Bigdeli, A., Zihayat, M., Bagheri, E.: Neural disentanglement of query difficulty and semantics. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. pp. 4264–4268 (2023)
39. Sarnikar, S., Zhang, Z., Zhao, J.L.: Query-performance prediction for effective query routing in domain-specific repositories. *Journal of the Association for Information Science and Technology* **65**(8), 1597–1614 (2014)
40. Shtok, A., Kurland, O., Carmel, D.: 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. pp. 259–266 (2010)
41. Shtok, A., Kurland, O., Carmel, D., Raiber, F., Markovits, G.: Predicting query performance by query-drift estimation. *ACM Transactions on Information Systems (TOIS)* **30**(2), 1–35 (2012)
42. Singh, A., Ganguly, D., Datta, S., McDonald, C.: 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. pp. 2486–2490 (2023)
43. Soboroff, I.: Overview of trec 2021. In: 30th Text REtrieval Conference. Gaithersburg, Maryland (2021)

44. Tao, Y., Wu, S.: 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. pp. 1891–1894 (2014)
45. Zamani, H., Croft, W.B., Culpepper, J.S.: Neural query performance prediction using weak supervision from multiple signals. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. pp. 105–114 (2018)
46. Zhou, Y., Croft, W.B.: Query performance prediction in web search environments. In: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. pp. 543–550 (2007)