Estimating Query Performance Through Rich Contextualized Query Representations

Sajad Ebrahimi	University of Guelph
Maryam Khodabakhsh	Shahrood University of Technology
Negar Arabzadeh	University of Waterloo
Ebrahim Bagheri	University of Toronto

Problem Definition

Dealing effectively with poorly-performing queries is a crucial issue in information retrieval systems.

Query Performance Prediction (QPP) models have been developed to estimate the performance of a system without the need for human-made relevance judgments.

In the post-retrieval we have:

- q: A query
- C: The collection of documents
- R: A retrieval method
- D_q: A ranked list of documents retrieved by R in response to query q

G. Faggioli et al., Query Performance Prediction for Neural IR: Are We There Yet?

Problem Definition

And our goal is to:

- Estimate the value of a given retrieval metric for query q.

$$\widehat{M_q} = \mu(q, D_q, C)$$

How can we evaluate the result?

- Models with a higher correlation between predicted values and actual values are better.

$$Quality(\mu) = correlation([\hat{M}(q1), \hat{M}(q2), ...], [M(q1), M(q2), ...])$$

Motivations and Foundational Ideas

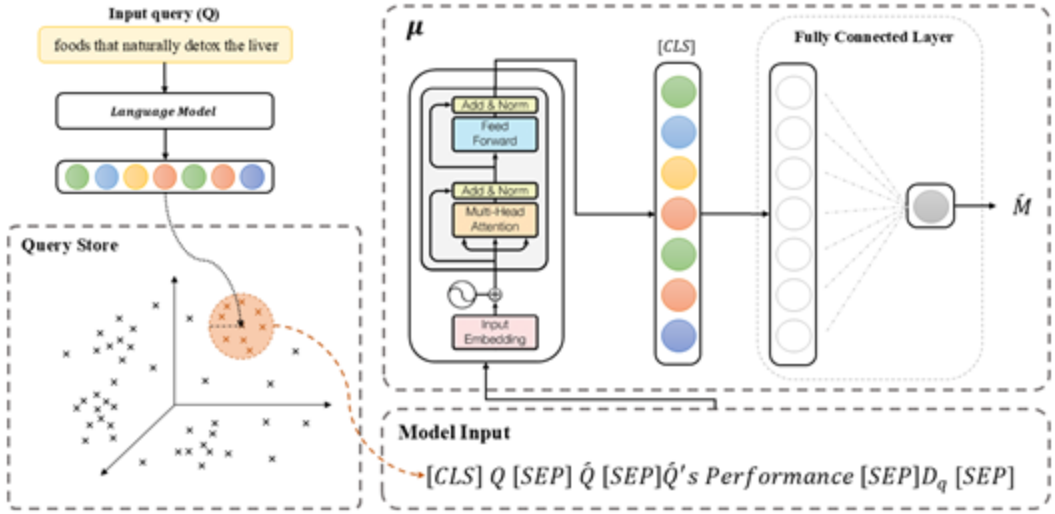
- Prior research has demonstrated the effectiveness of numerical signals in enhancing passage retrieval performance. This raises the question: Can similar numerical augmentations benefit Query Performance Prediction (QPP)?
- What types of numerical signals are most beneficial to inject into the input for QPP?
- Can we improve QPP by providing the model with a set of queries whose performance is already known, using their associated performance scores as numerical input?
- How to select a query from historical set to match to the given query?

Askari et al., Injecting the BM25 Score as Text Improves BERT-Based Re-rankers, ECIR 2023

Finding Nearest Neighbor queries for a given query:

Given Query		The most similar query from QueryStore	
id	text	id	text
190044	Foods to detox liver naturally	189691	Foods that naturally detox the liver
786674	What is prime rate in canada	481686	Prime rate canada definition

Proposed Approach



- The query q, the first document retrieved by the model in response to q, and the nearest neighbor query to q are concatenated.
- The concatenated sequence is input to a language model, followed by a linear layer.
- The loss function: $\ell(\widehat{M}_q, M(q, D_q)) = -w[M(q, D_q).log(\sigma(\widehat{M}_q)) + (1 - M(q, D_q).log(1 - \sigma(\widehat{M}_q)))]$

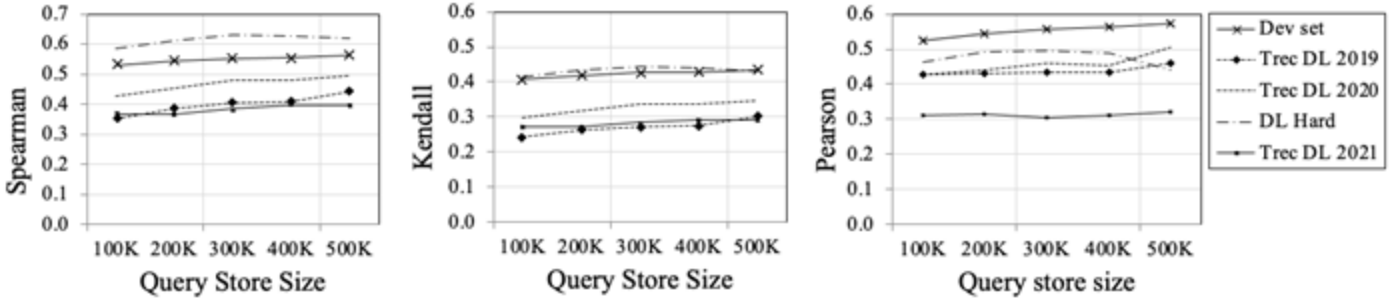
Results

	MS MARCO Dev			DL-Hard			2019			2020			2021		
	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$	$p-\rho$	$k-\tau$	$s-\rho$
Clarity	0.149	0.258	0.345	0.149	0.099	0.126	0.271	0.229	0.332	0.360	0.215	0.296	0.111	0.070	0.094
WIG	0.154	0.170	0.227	0.331	0.260	0.348	0.310	0.158	0.226	0.204	0.117	0.166	0.197	0.195	0.270
QF	0.170	0.210	0.264	0.210	0.164	0.217	0.295	0.240	0.340	0.358	0.266	0.366	0.132	0.101	0.142
NeuralQPP	0.193	0.171	0.227	0.173	0.111	0.134	0.289	0.159	0.224	0.248	0.129	0.179	0.134	0.221	0.188
$n(\sigma_{\%})$	0.221	0.217	0.284	0.195	0.120	0.147	0.371	0.256	0.377	0.480	0.329	0.478	0.269	0.169	0.256
RSD	0.310	0.337	0.447	0.362	0.322	0.469	0.460	0.262	0.394	0.426	0.364	0.508	0.256	0.224	0.340
SMV	0.311	0.271	0.357	0.375	0.269	0.408	0.495	0.289	0.440	0.450	**0.391**	**0.539**	0.252	0.192	0.278
NQC	0.315	0.272	0.358	0.384	0.288	0.417	0.466	0.267	0.399	0.464	0.294	0.423	0.271	0.201	0.292
UEF_{NQC}	0.316	0.303	0.398	0.359	0.319	0.463	0.507	0.293	0.432	**0.511**	0.347	0.476	0.272	0.223	0.327
NQA-QPP	0.451	0.364	0.475	0.386	0.297	0.418	0.348	0.164	0.255	0.507	0.347	0.496	0.258	0.185	0.265
BERT-QPP	0.517	0.400	0.520	0.404	0.345	0.472	0.491	0.289	0.412	0.467	0.364	0.448	0.262	0.237	0.34
qpp-BERT-PL	0.520	0.413	0.522	0.330	0.266	0.390	0.432	0.258	0.361	0.427	0.280	0.392	0.247	0.172	0.292
qpp-PRP	0.302	0.311	0.412	0.090	0.061	0.063	0.321	0.181	0.229	0.189	0.157	0.229	0.027	0.004	0.015
Ours	**0.555**	**0.421**	**0.544**	**0.434**	**0.412**	**0.508**	**0.519**	**0.318**	**0.459**	0.462	0.318	0.448	**0.322**	**0.266**	**0.359**

→ Language models can predict the performance of a query when provided with the performance of a similar query.
→ There is no single baseline that consistently achieves the best performance on DL 2019, DL 2020, and DL Hard.
→ Our approach demonstrates consistent behavior across all datasets and evaluation metrics.
→ Our model outperforms the baselines on four out of five test sets.

Results - Impact of Query Store Size

We randomly down-sampled the query store by only including 100k, 200k, 300k, 400k, and 500k from the training set.



→ Models that have been trained on smaller sets of queries are still effective.

Future works

- Applying the idea to the pre-retrieval
- Using the performance of the Nearest Neighbor queries without training
- Analysing the impact of various hyperparameters, such as:
    - Number of similar queries that are being added to the input
    - Size of D_q

Thank you!

Paper

Code

Presented by Sajad Ebrahimi: @SadjadEb sebrah05@uoguelph.ca

ECIR 2025