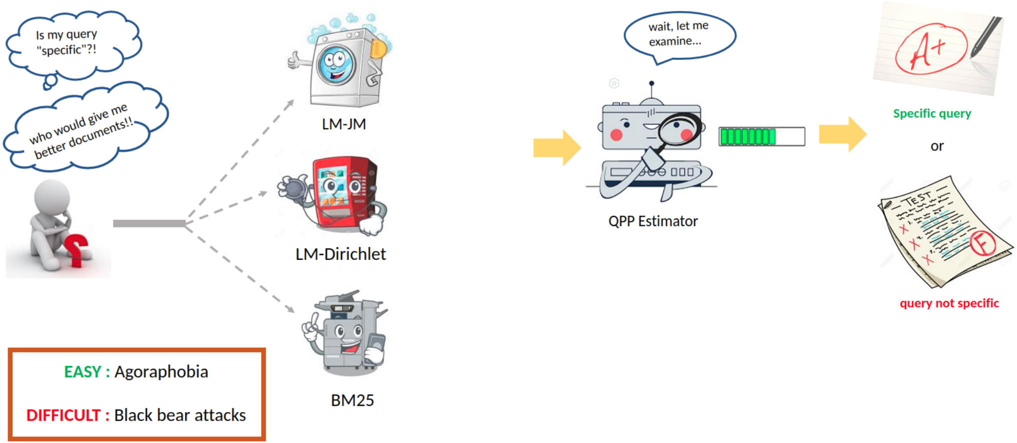
University of Glasgow | Terrier Team

Query Performance Prediction for Adaptive IR and RAG

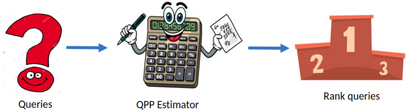
Debasis Ganguly
University of Glasgow

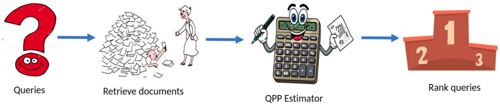
Table of Contents

What is Query Performance Prediction (QPP)?



QPP Estimator Types





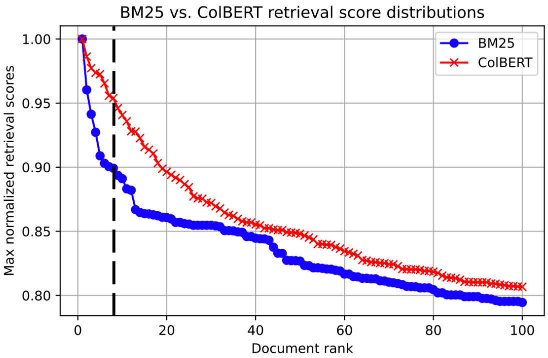
Pre-retrieval

- Input: only a query
- Agnostic of retrieval model
- Leverages collection statistics
- Functional form: $\phi : Q, \mapsto \mathbb{R}$

Post-retrieval

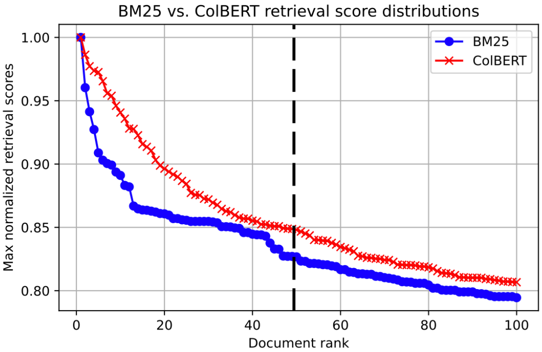
- Input: both a query and its top-retrieved list.
  - as obtained by a retrieval model θ.
- Prediction based on: How distinct is the top-k?
  - Distribution of retrieval scores, e.g., NQC.
  - Inter-document and collection-based measures, e.g., WIG, Clarity.
  - Robustness-based measures, e.g., UEF.
- Functional form: $\phi : Q, \underbrace{\boxed{L_k^{\theta}(Q)}}_{\text{top-retrieved}} \mapsto \mathbb{R}$

Score-based approaches



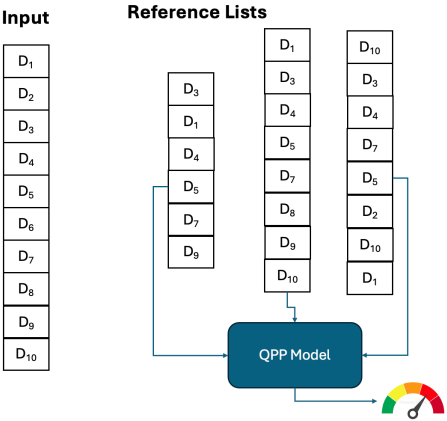
- Skewness of scores → relevant documents at the top
- A standard quantifier of skewness → Variance.
- Prediction depends on:
  - Number of documents considered (cut-off rank).
- Different models exhibit different score distribution.
- Skewness hypothesis may not be true.

Score-based approaches



- Skewness of scores → relevant documents at the top
- A standard quantifier of skewness → Variance.
- Prediction depends on:
  - Number of documents considered (cut-off rank).
- Different models exhibit different score distribution.
- Skewness hypothesis may not be always true.

Reference Lists



- More data helps!
- Aggregate predictors over more data.
- A simple way to get more inputs: randomly sample from $L_k^{\theta}(Q)$

UEF:

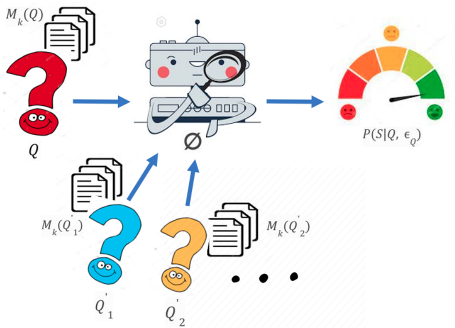
- Computes weighted average over random subsets.
- Weights: Stability of feedback models estimated for each list.

RLS:

- Takes a linear combination over the predictors for the reference lists and of the original input.

Query Variants to obtain Reference Lists

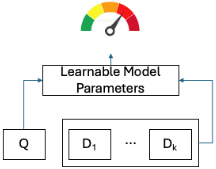
$$\phi(Q, L_k^{\theta}(Q)) \equiv \lambda\phi(Q, L_k^{\theta}(Q)) + (1-\lambda) \sum_{Q' \in \mathcal{E}_Q} \phi(Q', L_k^{\theta'}(Q'))\, \sigma(Q, Q')$$



- $\mathcal{E}_Q$: Queries with similar information needs - may be
  - manually generated (Zendel et. al., 2019),
  - automatically generated (Datta et. al., 2023),
  - retrieved from a query log (Tian et. al., 2025)
- The model θ may not be known, which means that a different model θ', such as BM25, can be used to obtain the retrieved lists for each variant.
- $\sigma(Q, Q')$: Measure of information need similarity – typically RBO of the top-retrieved.

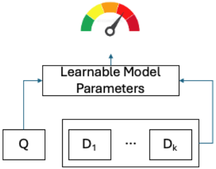
Supervised Approaches

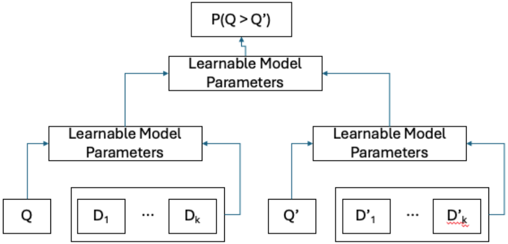
- $\phi : Q, L_k^\theta(Q) \mapsto \mathbb{R}$ – can be **learned from data**!
- **Pointwise**:
  $\mathcal{L}(\phi) = \sum_{Q \in \mathcal{Q}} (\theta(Q, L_k^\theta(Q)) - \mathcal{M}(L_k^\theta(Q), R(Q)))^2$
  - $\mathcal{M}$ is an IR metric
  - $R(Q)$ - a set of relevance assessments for Q
  - $\mathcal{Q}$: training set of queries.



Supervised Approaches

- $\phi : Q, L_k^\theta(Q) \mapsto \mathbb{R}$ – can be **learned from data**!
- **Pointwise**:
  $\mathcal{L}(\phi) = \sum_{Q \in \mathcal{Q}} (\theta(Q, L_k^\theta(Q)) - \mathcal{M}(L_k^\theta(Q), R(Q)))^2$
  - $\mathcal{M}$ is an IR metric
  - $R(Q)$ - a set of relevance assessments for Q
  - $\mathcal{Q}$: training set of queries.
- **Pairwise**: Learn to compare between two queries.
- $\mathcal{L}(\phi) = \sum_{(Q,Q') \in \mathcal{Q} \times \mathcal{Q}} \max(0, 1 - \mathrm{sgn}(y(Q) - y(Q'))\,(\hat{y}(Q;\phi) - \hat{y}(Q';\phi)))$
  - $y(Q) \equiv \mathcal{M}(L_k^\theta(Q), R(Q)))$: ground-truth evaluation measure
  - $\hat{y}(Q;\phi) = \phi(Q, L_k^\theta(Q))$ - predicted evaluation measure





Late vs. Early Interaction

- Parameterized interactions between queries and documents.
- Bi-encoder (least parameters), cross-encoder (most parameters) or late interaction (good compromise).

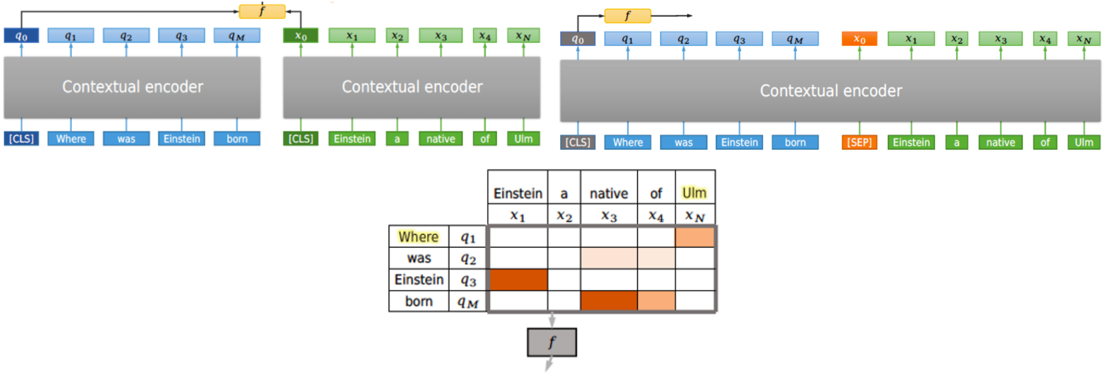
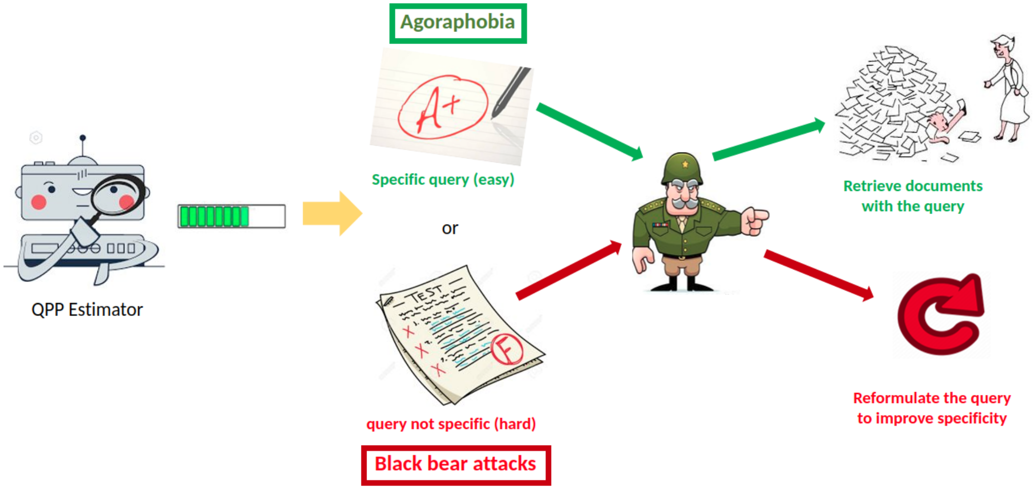


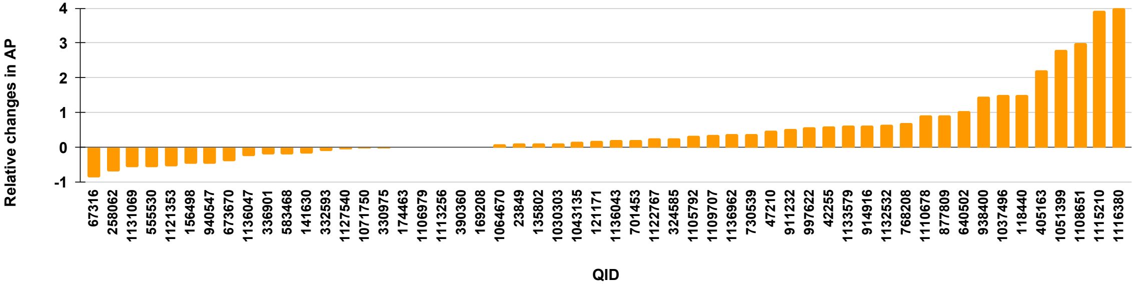
Table of Contents

What's the use of QPP?

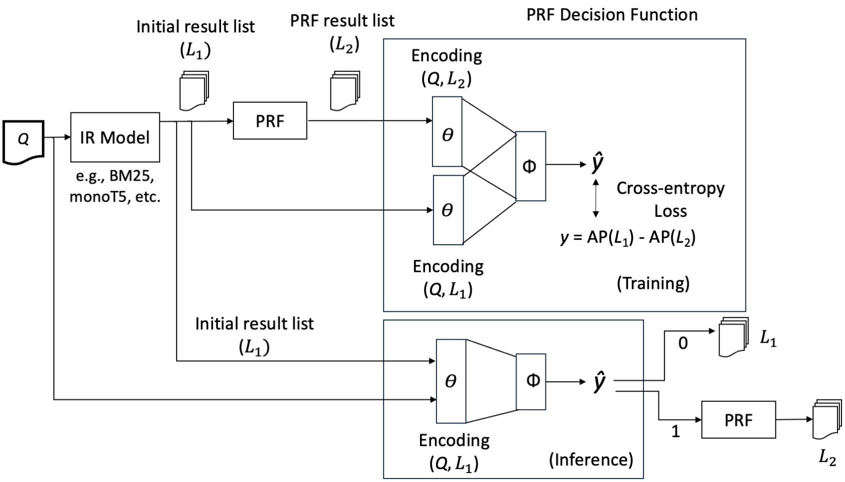


QPP for Adaptive IR

- Multi-stage ranking models → retrieve-rerank pipeline
- Stages with increasing computational complexity
  - BM25 » RM3, BM25 » Mono-T5, Contriever-E2E » Mono-T5 » Duo-T5 etc.
- Not all queries are benefited by the subsequent stages.

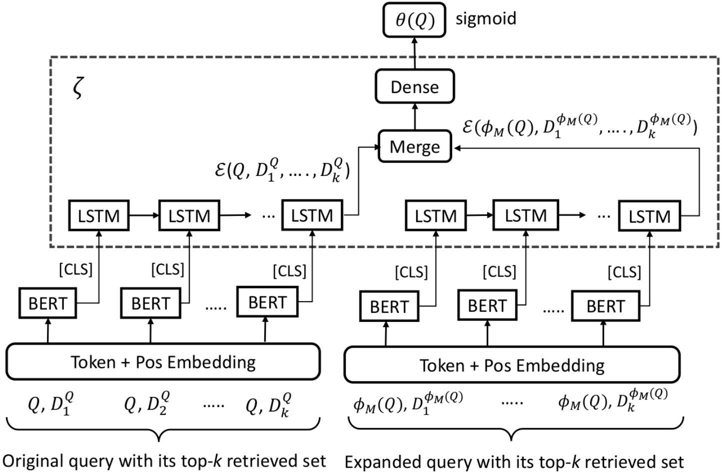


Classifier to select between two lists (Datta et. al. ECIR'24)



- **Training**: relevance assessments to decide which list is better.
- Inference: **Locality hypothesis** - Similar topics would behave similarly.

Model Architecture (Datta et. al. ECIR'24)



- Encodes sequence of documents with LSTMs.
- Cross-encoders not suitable to model $\langle D_1, \ldots, D_k \rangle$ when document sizes are relatively large.
- **Soft selection**: The Sigmoid $p : 1 - p$ used as weights to combine the two lists.

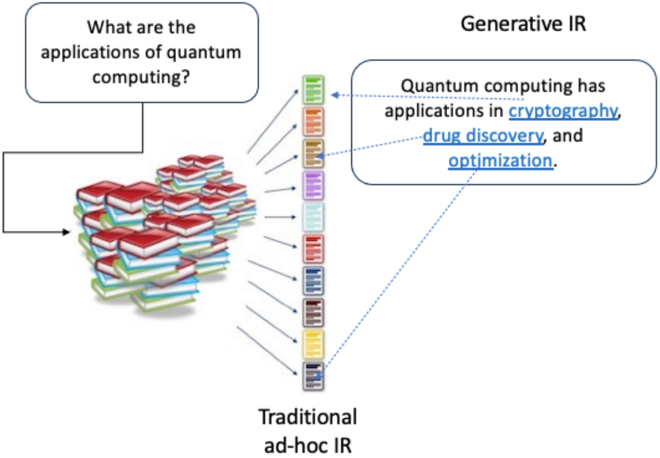
Adaptive IR works

	Methods	BM25 (ϕ: RLM)			BM25 (ϕ: GRF)			BM25 (ϕ: ColBERT-PRF)		
		Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10
Baselines	No PRF	N/A	0.3766	0.5022	N/A	0.3766	0.5022	N/A	0.3766	0.5022
	PRF	N/A	0.4321	0.5134	N/A	0.4883	0.6226	N/A	0.4514	0.6067
	R2F2	N/A	0.4381	0.5140	N/A	0.5094	0.6332	N/A	0.4968	0.6184
	QPP-SRF	0.7835	0.4400	0.5152	0.7844	0.5321	0.6667	0.7742	0.5238	0.6400
	TD2F	0.7611	0.4392	0.5135	0.7580	0.4579	0.5900	0.7642	0.4910	0.6038
	LR-SRF	0.7842	0.4411	0.5154	0.7784	0.5107	0.6512	0.7854	0.5254	0.6414
Ours	Deep-SRF-BERT	**0.8081***	0.4705	0.5374	**0.8093***	0.5654	0.6821	**0.8165***	0.5631	0.6765
	Deep-SRF-BERT-R2F2		**0.4961**	**0.5486**		**0.5730**	**0.6839**		**0.5785**	**0.6873**
	Oracle	1.0000	0.5038	0.5528	1.0000	0.5876	0.6941	1.0000	0.5820	0.6936

	Methods	MonoT5 (ϕ: RLM)			MonoT5 (ϕ: GRF)			MonoT5 (ϕ: ColBERT-PRF)		
		Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10
Baselines	No PRF	N/A	0.5062	0.6451	N/A	0.5062	0.6451	N/A	0.5062	0.6451
	PRF	N/A	0.5081	0.6463	N/A	0.5200	0.6487	N/A	0.5297	0.6491
	R2F2	N/A	0.5112	0.6484	N/A	0.5241	0.6494	N/A	0.5324	0.6502
	QPP-SRF	0.7963	0.5189	0.6559	0.7871	0.5313	0.6604	0.7900	0.5419	0.6673
	TD2F	0.7789	0.5071	0.6453	0.7670	0.4991	0.6403	0.7612	0.5179	0.5986
	LR-SRF	0.7958	0.5180	0.6543	0.7980	0.5422	0.6628	0.7928	0.5500	0.6654
Ours	Deep-SRF-BERT	**0.8152***	0.5306	0.6640	**0.8160***	0.5529	0.6694	**0.8067***	0.5624	0.6733
	Deep-SRF-BERT-R2F2		**0.5317**	**0.6659**		**0.5607**	**0.6719**		**0.5711**	**0.6746**
	Oracle	1.0000	0.5416	0.6786	1.0000	0.5722	0.6803	1.0000	0.5801	0.6821

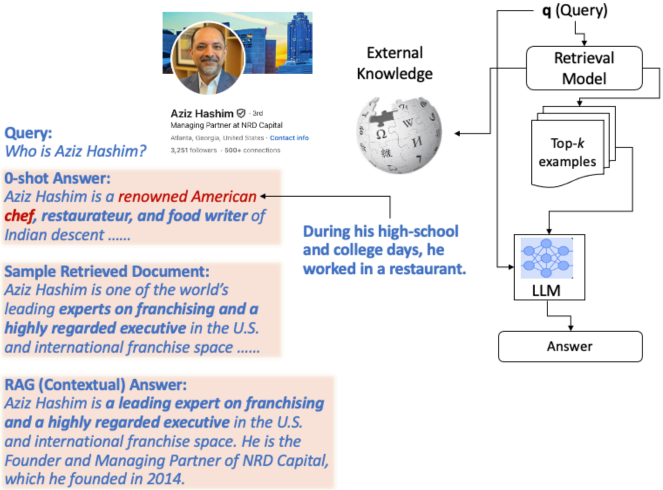
Table of Contents

Generative IR



- Consuming relevant information as a ranked list of documents → more cognitive effort by users.
- A single generated answer with links to more information (if reqd.) → reduces user effort.

The role of retrieved information in RAG



- Zero-shot answers can contain misinformation.
- Conditional generation provides correct and more informative answers.

Adaptive RAG (Parry et al., 2024)

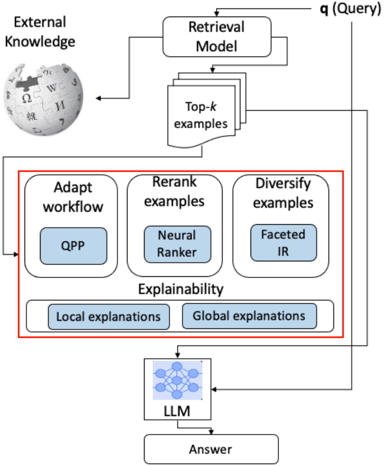
Query:
Who is Aziz Hashim?

0-shot Answer:
*Aziz Hashim is a renowned American **chef, restaurateur, and food writer** of Indian descent ......*

During his high-school and college days, he worked in a restaurant.

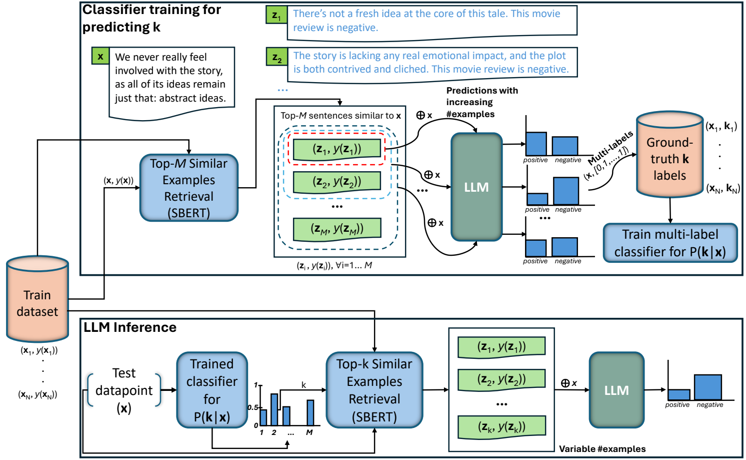
Sample Retrieved Document:
*Aziz Hashim is one of the world's leading **experts on franchising and a highly regarded executive** in the U.S. and international franchise space ......*

RAG (Contextual) Answer:
*Aziz Hashim is **a leading expert on franchising and a highly regarded executive** in the U.S. and international franchise space. He is the Founder and Managing Partner of NRD Capital, which he founded in 2014.*



- QPP → **utility** of a context
- Maybe applied to adjust the hyper-parameters of RAG, e.g., the number of documents etc.

Adapt RAG Context Size (Chandra et al., ECIR'25 - Best Paper)



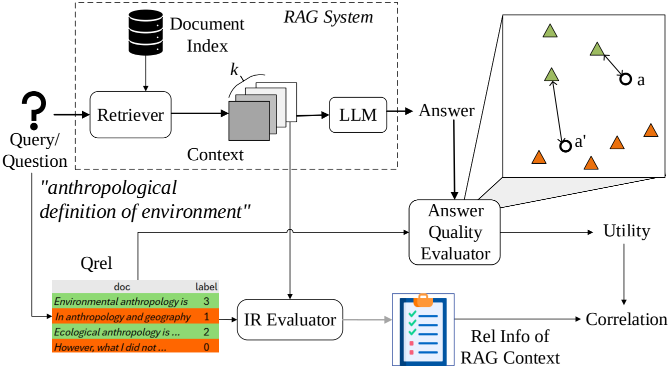
- Static context size:
- $P(y|\mathbf{x},k) = f(\mathbf{x},\mathcal{N}_k(\mathbf{x});\phi_{\text{LLM}})$
- Dynamic context size (depends on input):
- $P(y|\mathbf{x},\kappa) = f(\mathbf{x},\mathcal{N}_{\kappa(\mathbf{x})}(\mathbf{x});\phi_{\text{LLM}})$
- $\kappa : \mathbf{x} \mapsto \{0,\ldots,M\}$
- Locality hypothesis: Topically similar questions (inputs) should have similar optimal context sizes.
  - *M*: upper bound of context size
- Training: Learn classifier with a downstream performance measure.
- Inference: context size is set to the integer predicted by the classifier.

Adapting context size helps!

		RAG setup (w/o Labels)			ICL setup (w/ Labels)		
Dataset	0-shot	FICL	AICL(E)	AICL*	FICL	AICL(E)	AICL*
SST2	.8914	.7339	**.9119**	.9610	.9252	.9300	.9863
TREC	.3526	.4287	**.4752**	.4922	.6192	.7196	.9313
CoLA	.2558	.2469	**.2679**	.7937	.6433	.6601	.9413
RTE	**.6741**	.6144	.6688	.8655	.7240	.7415	.9234

- Adaptive ICL with neighborhood homogeneity (AICL+E) outperforms Fixed ICL.
  - Improves results both for labeled and unlabeled data.
- Further improvement (see oracle results).

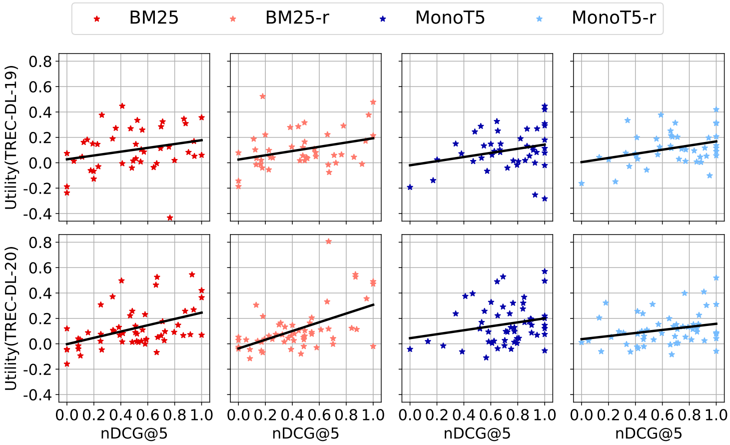
Utility of RAG contexts (Tian et al., ECIR'25)



- Some contexts are more useful than others.
- Only some lead to gains in performance measure w.r.t. zero-shot
- Utility: Relative gain in downstream performance
- Gains more important when 0-shot performance is low (similar to IR performance).

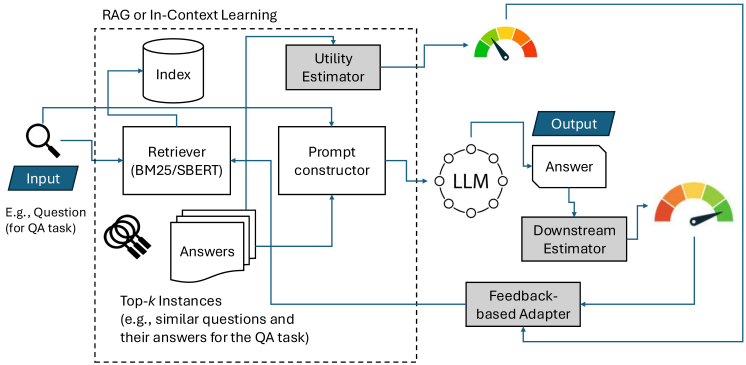
- Performance measure: semantic similarity with judged relevant documents (Arabzadeh et al., ECIR'24)

Is Utility Correlated with Relevance?



- Is utility mainly a function of relevance, or there is something else to it?
- Positive but **small correlation**.
- Computationally expensive rankers don't add up much to RAG performance.

A Generic Adaptive RAG workflow



- **Utility Estimator** or **Retriever-PP**: QPP over RAG context
  - Not in terms of relevance.
  - But in terms of **utility**.
- **Downstream Estimator** or **Generator-PP**: Predict performance for the downstream answer.
  - Pre-generation (predict performance w/o generation)
  - Post-generation (predict performance *after generation*).
- **Feedback**: Feedback from these predictors can then be used to modify a RAG system.

Some preliminary results from work-in-progress

- RPP: Just apply a QPP method on the input and the RAG context.
- GPP: Treat the generated answer as a query, retrieving from the collection. Execute QPP on this list.
- Pre-generation GPP ≈ Pre-retieval QPP (most challenging).

					Pre-CG Predictions								Post-CG Predictions			
			QPP		RPP				GPP				GPP			
					w/o posteriors		w/ posteriors		w/o posteriors		w/ posteriors		w/o posteriors		w/ posteriors	
θ_R	Type	Method	DL'19	DL'20	DL'19	DL'20	DL'19	DL'20	DL'19	DL'20	DL'19	DL'20	DL'19	DL'20	DL'19	DL'20
BM25	Unsupervised (RSV)	NQC	.1777	*.2988	.0365	*.2131	.0410	*.2551	.1096	.1530	.1473	*.2006	*.3621	*.2439	*.5061	*.3096
		UEF	.1577	*.3269	.0565	*.2341	.0543	*.2607	.1096	.1391	.1606	*.1978	***.3643**	*.2411	*.5017	*.3082
		RSD	.1399	*.2876	.0432	*.2271	.0520	*.2509	.1074	.1530	.1517	*.2020	*.3621	*.2439	*.4928	*.3082
	Unsupervised (EMB)	QPP-Dense	*.2776	*.3297	***.3200**	***.3040**	**.1340**	***.4018**	***.2602**	***.3068**	***.3178**	***.4270**	*.2536	***.3110**	*.5127	*.4326
		A-Ratio	*.3376	*.3788	.2004	*.2257	.0100	*.3389	.0388	.0594	*.2647	*.3403	.1805	*.2145	***.5637**	***.4507**
	Supervised	QPP-BERT	*.3531	***.4195**	.0720	*.2690	.1074	*.3110	.0210	.1209	.1118	*.2271	*.3178	*.2565	*.5194	*.3725
		QPP-BERT(QV)	***.3598**	*.4167	.0853	*.2774	.0919	*.3040	-.0166	.1125	.1008	*.1838	*.2890	*.2299	*.4839	*.3515

- Existing QPP approaches work fairly well.
- Possible scope of further improvements with additional features, such as coherence.

Ways to adapt a RAG system

- $\downarrow$RPP $\rightarrow$ improve the retriever.
  - More computationally expensive ranker.
  - Increase the context size.
  - ...
- $\downarrow$GPP $\rightarrow$ Improve the generator.
  - More computationally expensive generator (LLM with more parameters).
  - Reason (Chain of Thoughts).
  - ...

Thank you!

Questions?