Fine Tuning vs. Retrieval Augmented Generation for Less Popular Knowledge

Heydar Soudani[1], Evangelos Kanoulas[2], Faegheh Hasibi[1]

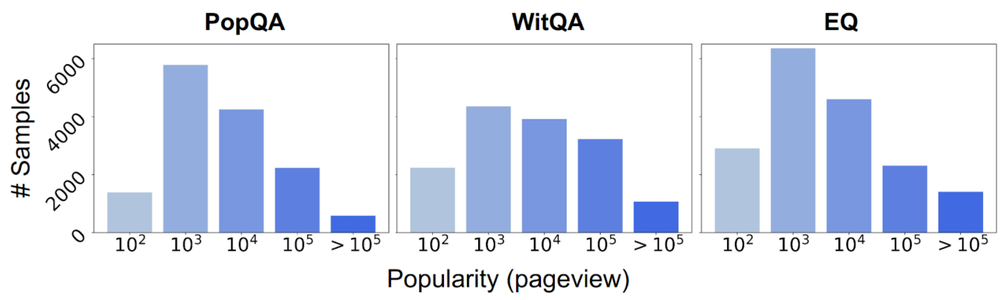
Radboud University[1] University of Amsterdam[2]

1 Introduction

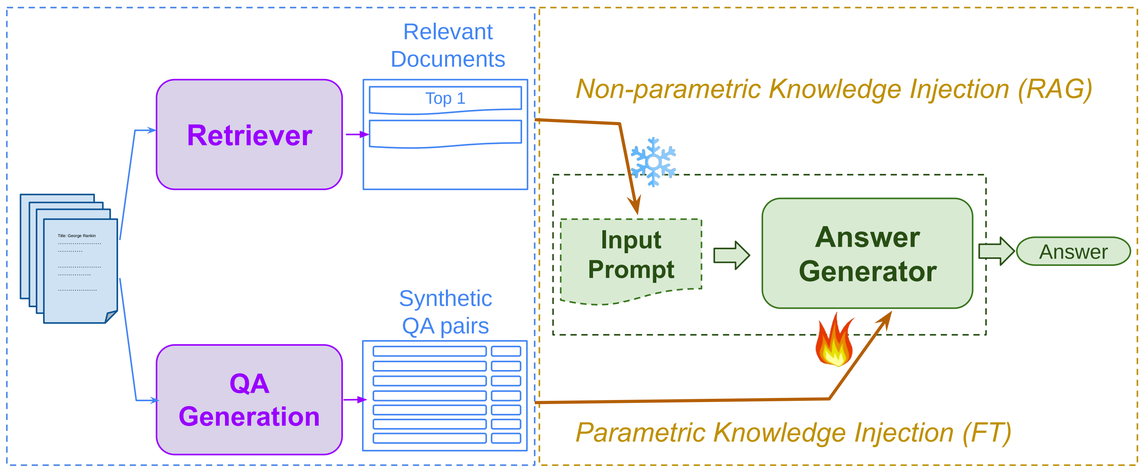
- **Challenge:** LLMs struggle to memorize less popular or domain-specific concepts
- **RQ1:** How does RAG compare to fine-tuning for question answering over less popular factual knowledge, and which factors affect their performance?
- **RQ2:** Can we avoid the cost of fine-tuning by developing an advanced RAG approach that surpass the performance of a fine-tuned LM with RAG?
- **Focus:** Factual knowledge, information that describes particular attributes or characteristics of target entities

(Kathy Saltzman, Occupation, Politician) → **Q:** What is the occupation of Kathy Saltzman?
Subject Relationship Object **A:** Politician

- **Evaluation Sets:** Wikipedia-based QA datasets
- Considering WP pageviews as the proxy for measuring query's popularity

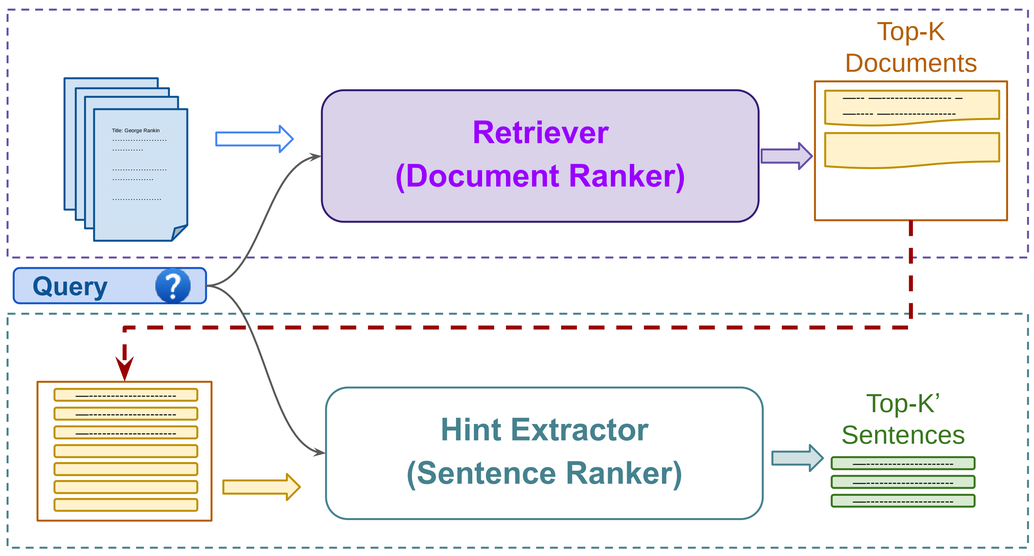


2 Evaluation Framework



(1) -FT-RAG: the vanilla LM without retrieved documents
(2) -FT+RAG: the vanilla LM with retrieved documents
(3) +FT-RAG: the fine-tuned LM without retrieved documents
(4) +FT+RAG: the fine-tuned LM with retrieved documents

3 Stimulus RAG

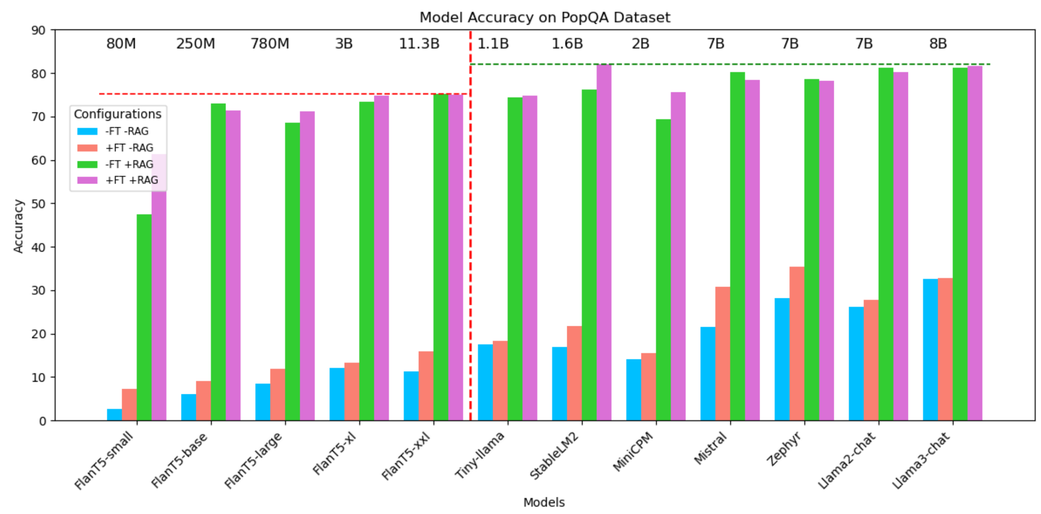


4 Results

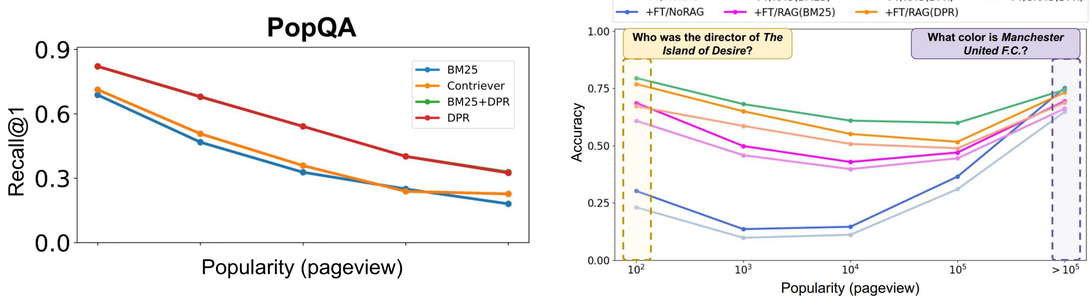
Evaluation Framework

1) Fine-tuning Methods: Full FT vs. PEFT
2) Data Augmentation Methods: End2End vs. Prompting
3) LLMs' sizes: From 80M to 11.3B &
   LLMs' types: Decoder-Encoder vs Decoder-only
4) Retrieval Models in RAG

		PopQA		EQ	
FT	**QA**	**+FT-RAG**	**+FT+RAG**	**+FT-RAG**	**+FT+RAG**
FlanT5-base		6.01	73.08	6.07	53.92
PEFT	E2E	7.53	70.34	10.98	51.30
PEFT	Prompt	$9.11^{(a,b,c)}$	$71.34^{(a,b,c,d)}$	$12.98^{(a,b,c,d)}$	$57.63^{(a,b,c,d)}$
Full	E2E	7.42	44.76	10.91	31.22
Full	Prompt	10.06	51.80	17.36	54.07
FlanT5-large		8.44	68.56	16.94	52.64
PEFT	E2E	8.69	67.47	15.33	53.25
PEFT	Prompt	$11.24^{(a,b,d)}$	$71.27^{(a,b,c,d)}$	$18.17^{(a,b,c)}$	$60.08^{(a,b,c,d)}$
Full	E2E	11.75	27.31	14.79	23.17
Full	Prompt	13.60	68.18	18.22	57.37



Examining QA performance across popularity buckets



Stimulus RAG (SRAG)

Model		**-FT+RAG (3D)**	**+FT+RAG (3D)**	**SRAG (S)**	**SRAG (D)**
		PopQA			
FlanT5-base	DPR	56.67	53.46	**57.77**$^{(a,b)}$	57.67$^{(a,b)}$
	Ideal	73.06	72.02	75.08$^{(a,b)}$	**75.29**$^{(a,b)}$
StableLM2	DPR	63.98	65.33	65.48$^{(a)}$	**66.01**$^{(a)}$
	Ideal	80.82	82.98	82.83$^{(a)}$	**83.18**$^{(a)}$
Mistral	DPR	65.22	63.63	65.84$^{(a,b)}$	**66.04**$^{(a,b)}$
	Ideal	81.58	80.30	81.88$^{(b)}$	**82.27**$^{(a,b)}$
Llama3	DPR	66.66	66.61	**67.22**	67.21
	Ideal	**82.58**	**82.58**	82.42	81.60

5 Takeaways

- RAG significantly outperforms fine-tuning alone
- Fine-tuned LMs with RAG either outperform or match vanilla LMs with RAG
- RAG is particularly beneficial for less popular entities
- Improvements from fine-tuning are not influenced by entity popularity
- Advanced RAG systems can achieve better accuracy than fine-tuning, avoiding its complexities and resource demands

heydar.soudani@ru.nl