Fine Tuning vs. Retrieval Augmented Generation for Less Popular Knowledge

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

[1]Radboud University

[2]University of Amsterdam

SIGIR-AP 2024

Factual Knowledge

LM parameters encode a wealth of factual information

What is the highest mountain in Japan?

GPT-4

The highest mountain in Japan is **Mount Fuji**.

Less-popular Knowledge

LLMs struggle to memorize less popular or domain-specific concepts

Domain Specific Example: Bol.com

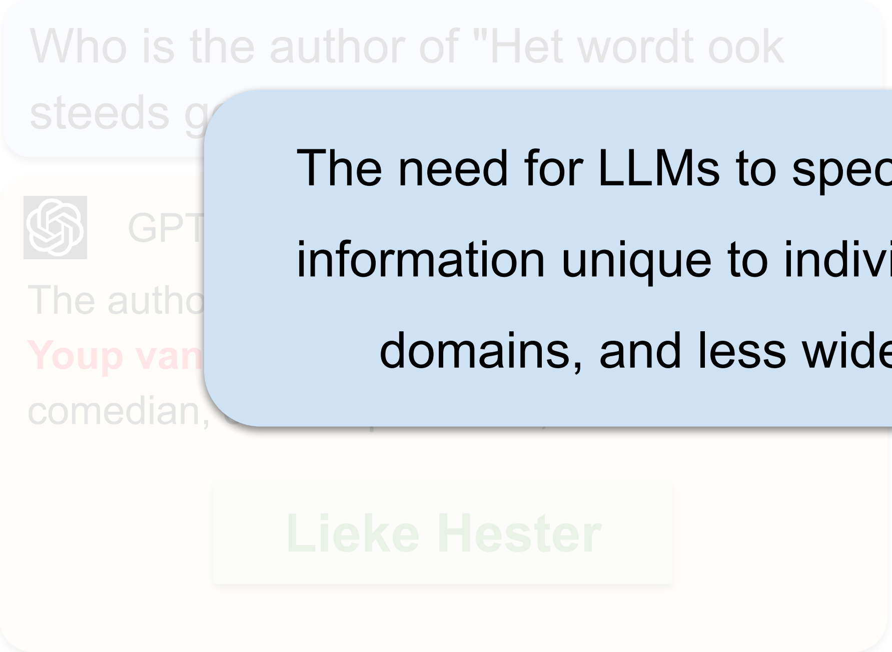
Query: Who is the author of "Het wordt ook steeds gekker"?

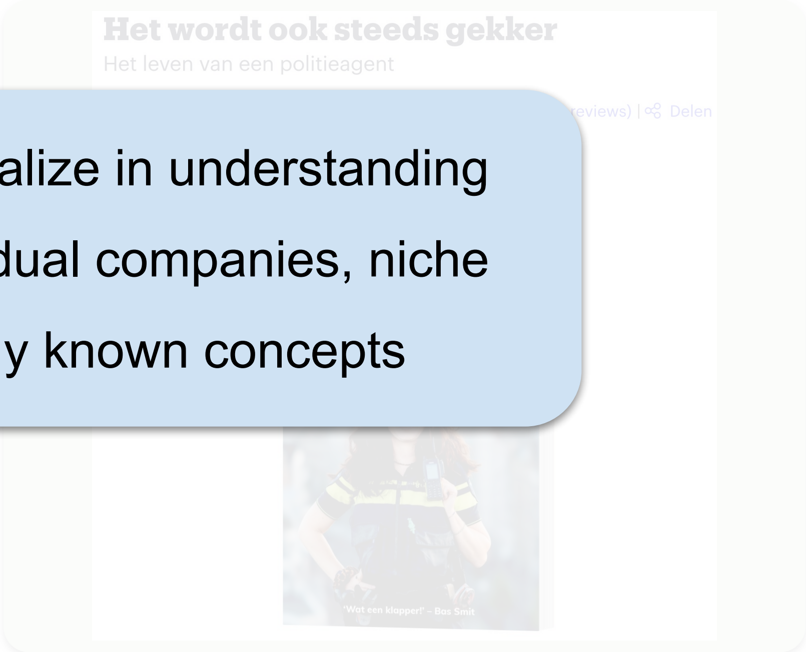
GPT-4o Answer: Youp van't Hek

✓ **Correct Answer:** Lieke Hester

Less-popular Knowledge

LLMs struggle to memorize less popular or domain-specific concepts

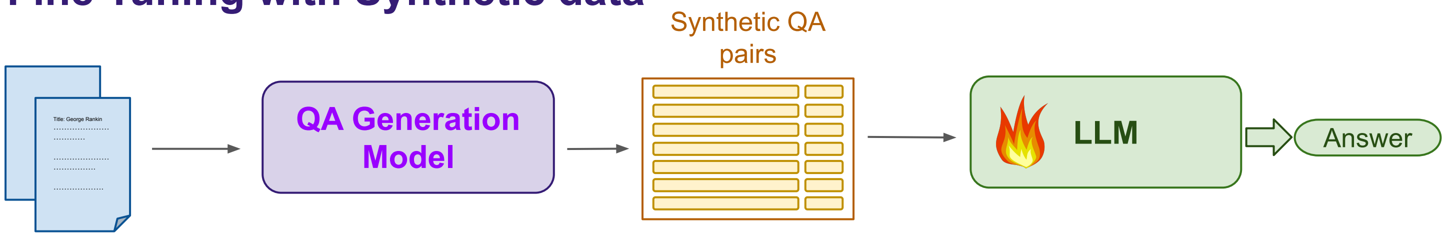




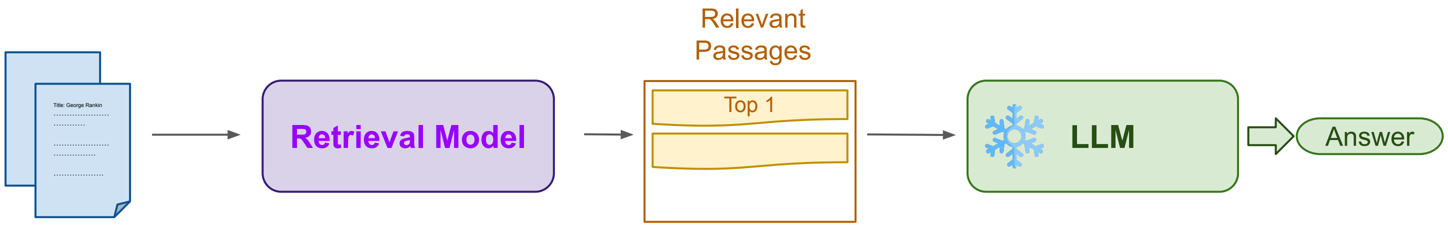
The need for LLMs to specialize in understanding information unique to individual companies, niche domains, and less widely known concepts

LLM Adaptation

Fine Tuning with Synthetic data



Retrieval Augmented Generation (RAG)



Research Questions

RQ1: How does RAG compare to fine-tuning (with synthetic data) for question answering over less popular factual knowledge?

❖ Which factors affect their performance?

1) Fine-tuning Methods: Full FT vs. PEFT
2) Data Augmentation
3) LLM type and size
4) Retrieval Model

Task Definition

How can we assess the memorized knowledge in a language model?

- **Focus:** Factual knowledge
  - Information that describes particular attributes of target entities
  - Triplet format

    (Kathy Saltzman, Occupation, Politician)

    Subject — Relationship — Object
  - The model successfully memorizes the knowledge if it can generate the correct object when given the subject and the relationship

Task Definition

- **Task:** Open-domain QA
  - The question incorporates the subject and the relationship
  - The answer corresponds to the object

(Kathy Saltzman, Occupation, Politician)

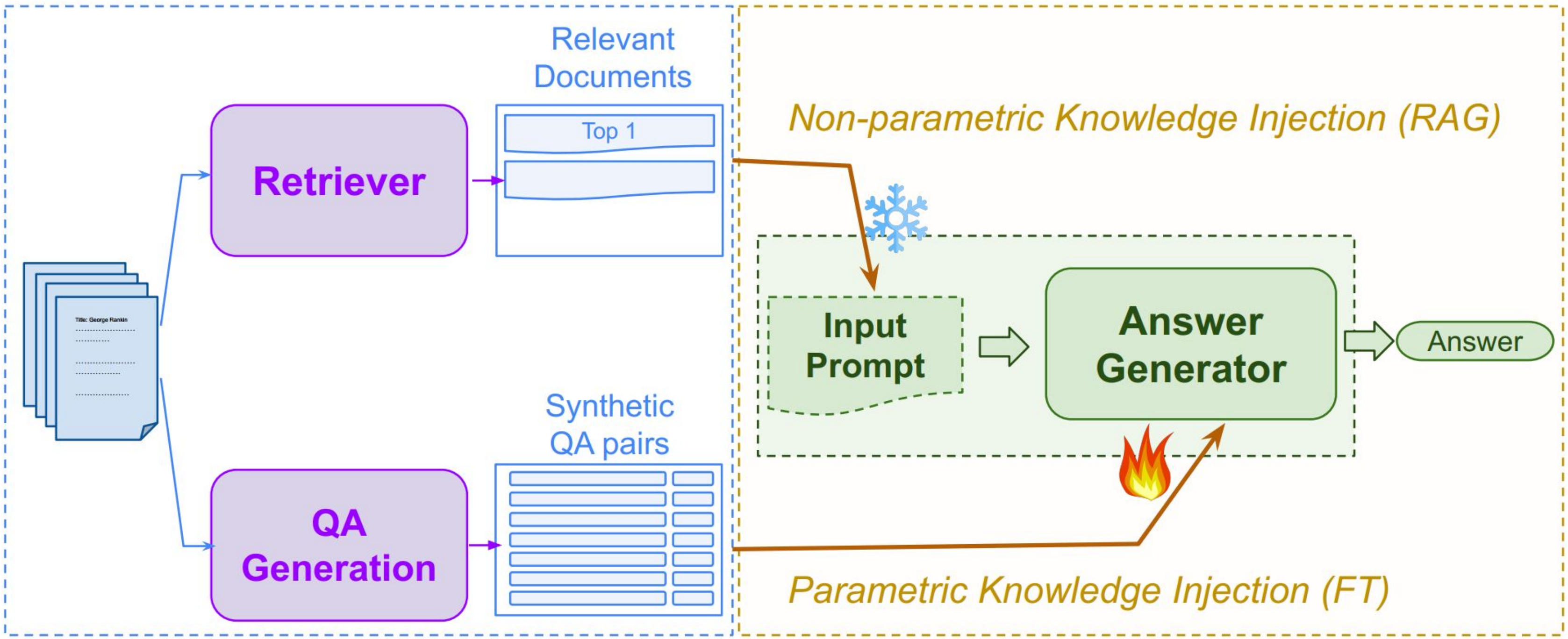
Subject Relationship Object

Q: What is the occupation of Kathy Saltzman?

A: Politician

- **Popularity:**
  - Wikipedia pageviews

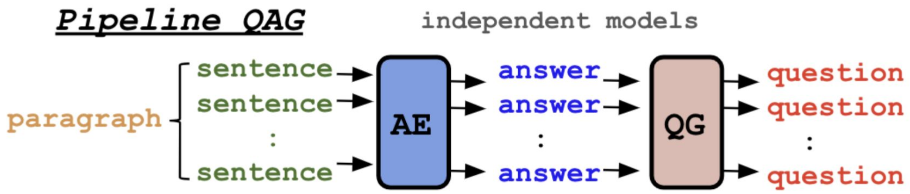
Methodology: Evaluation Framework

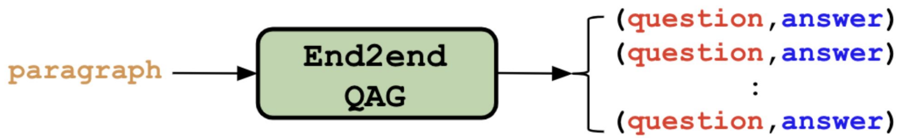


Methodology: Data Generation

End2End

- Based on pipeline method
  - Answer Extraction
  - Question Generation
- Combined them in one step





Methodology: Data Generation

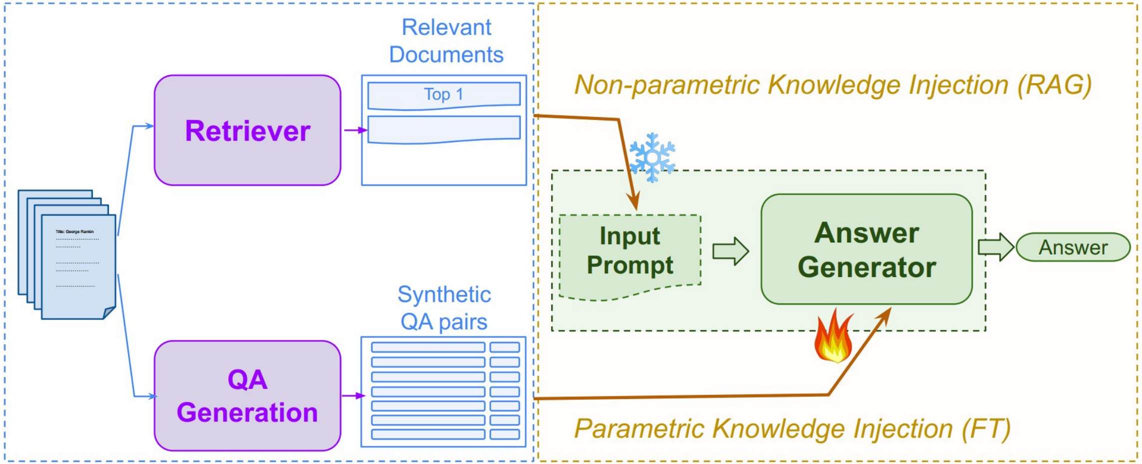
End2End

- Based on pipeline method
  - Answer Extraction
  - Question Generation
- Combined them in one step

Prompting

- Use an instruction-tuned language model
- CoT reasoning following two steps **explicitly**

Methodology: Configurations



(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

Results: Fine-tuning Methods

For LMs with less than 2 billion parameters, full FT is more effective than PEFT in the downstream task.

		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

Results: Fine-tuning Methods

PEFT preserves the reasoning ability of LMs (needed for RAG)

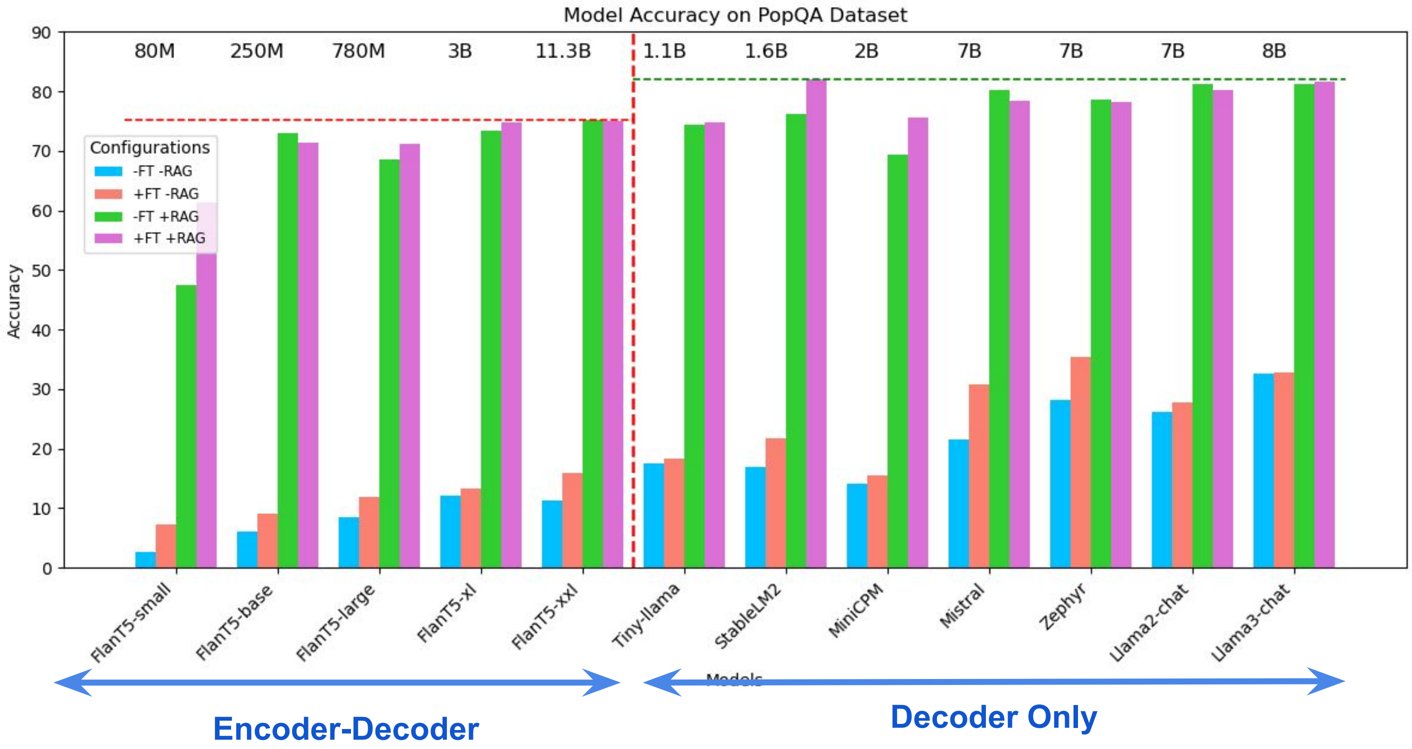
		PopQA		EQ	
FT	QA	+FT-RAG	+FT+RAG	+FT-RAG	+FT+RAG
FlanT5-base		6.01	73.08	6.07	53.92
PEFT	E2E	7.53	70.34	10.98	51.30
PEFT	Prompt	9.11(a,b,c)	71.34(a,b,c,d)	12.98(a,b,c,d)	57.63(a,b,c,d)
Full	E2E	7.42	44.76	10.91	31.22
Full	Prompt	10.06	51.80	17.36	54.07
FlanT5-large		8.44	68.56	16.94	52.64
PEFT	E2E	8.69	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

Results: Data Generation

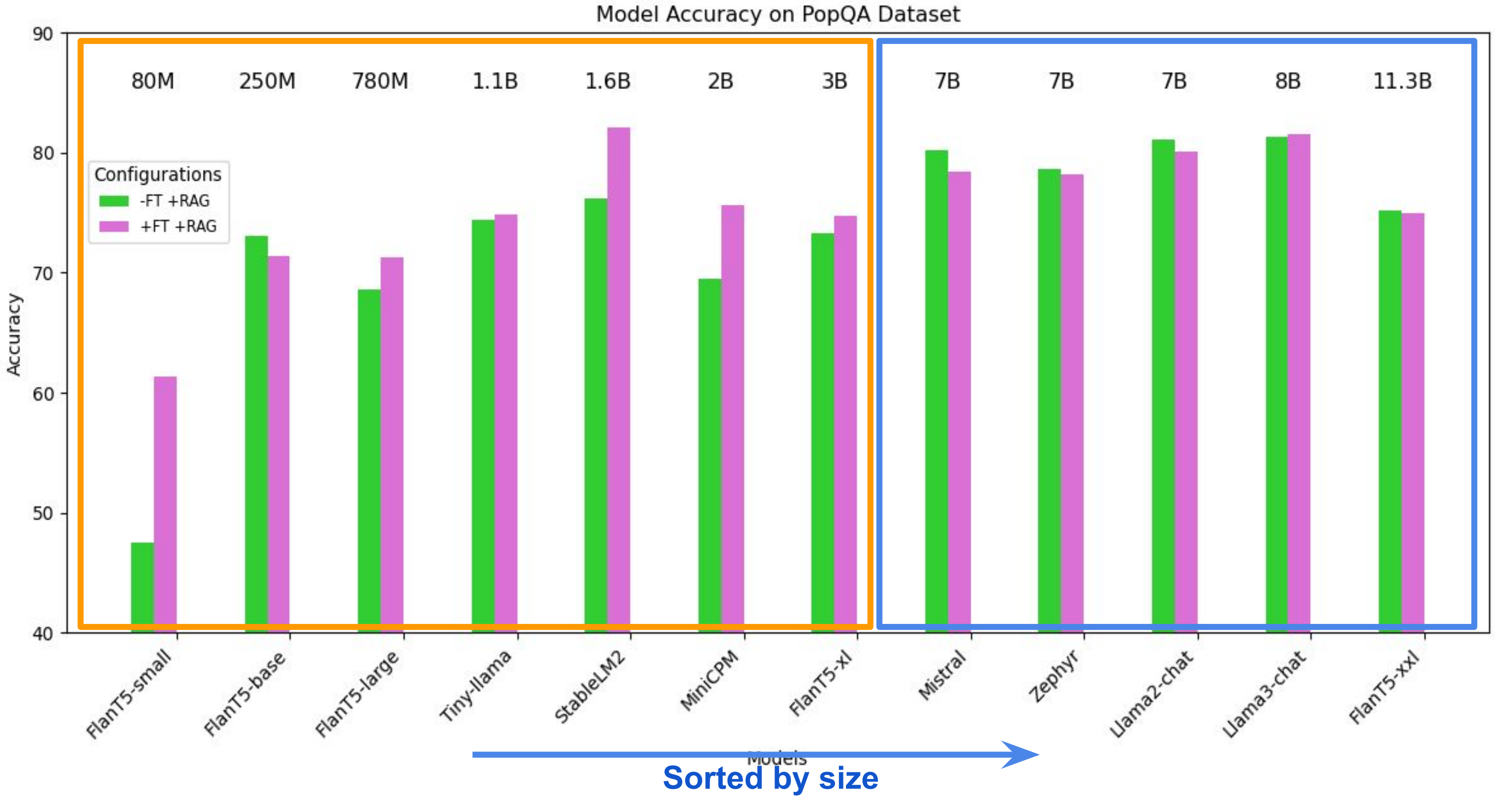
Quality vs. Quantity

		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

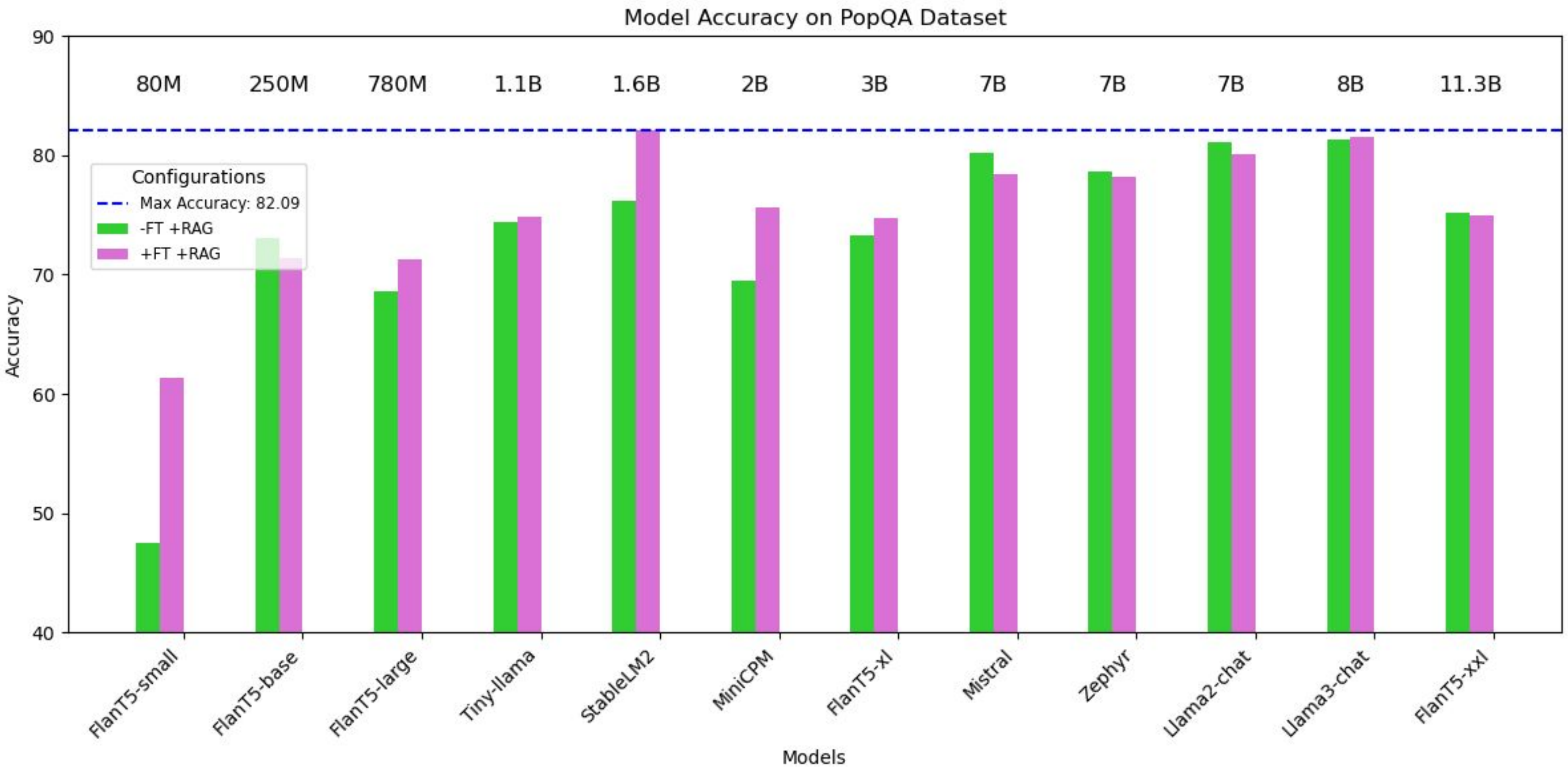
Results: LM type



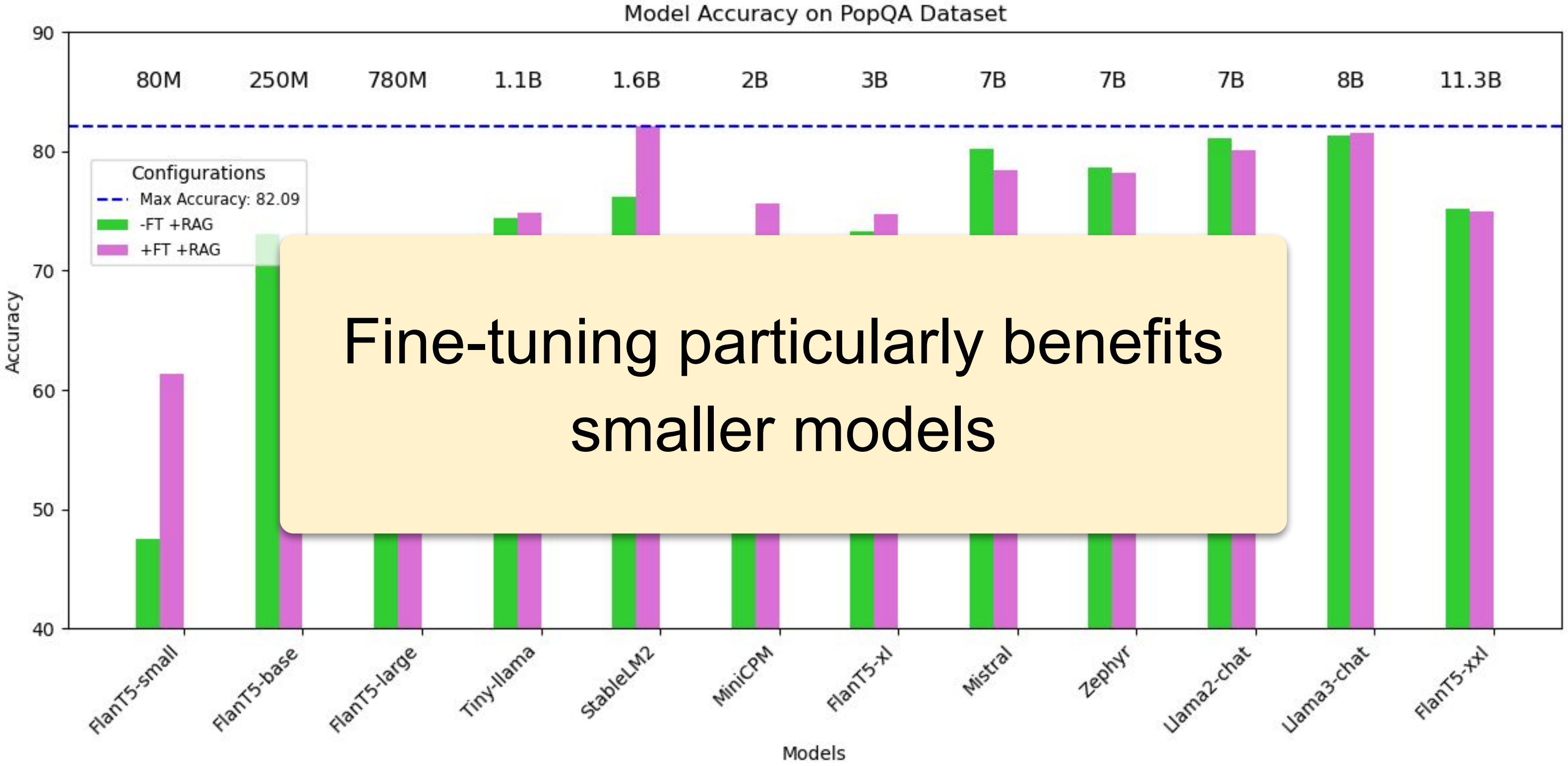
Results: LM size



Results: LM size

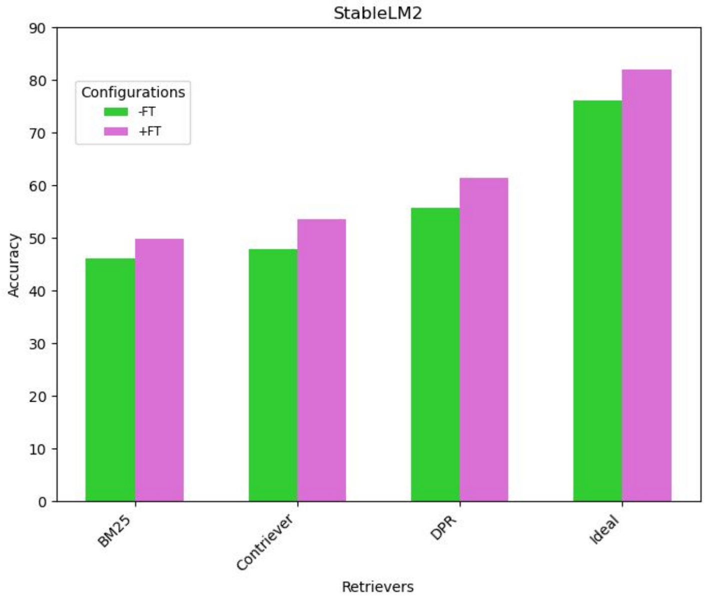


Results: LM size



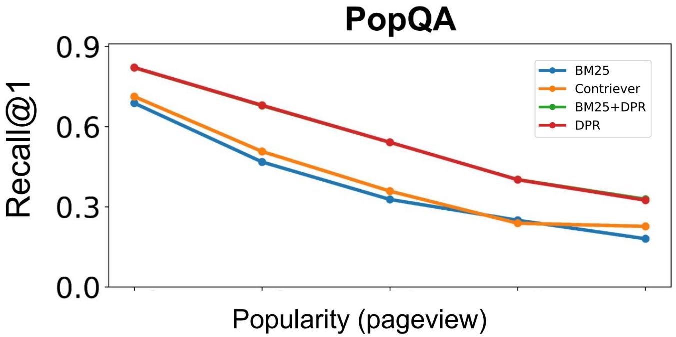
Fine-tuning particularly benefits smaller models

Results: Retrieval Model

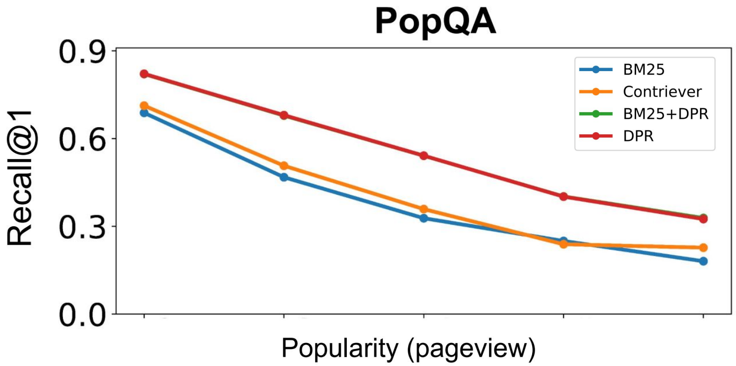


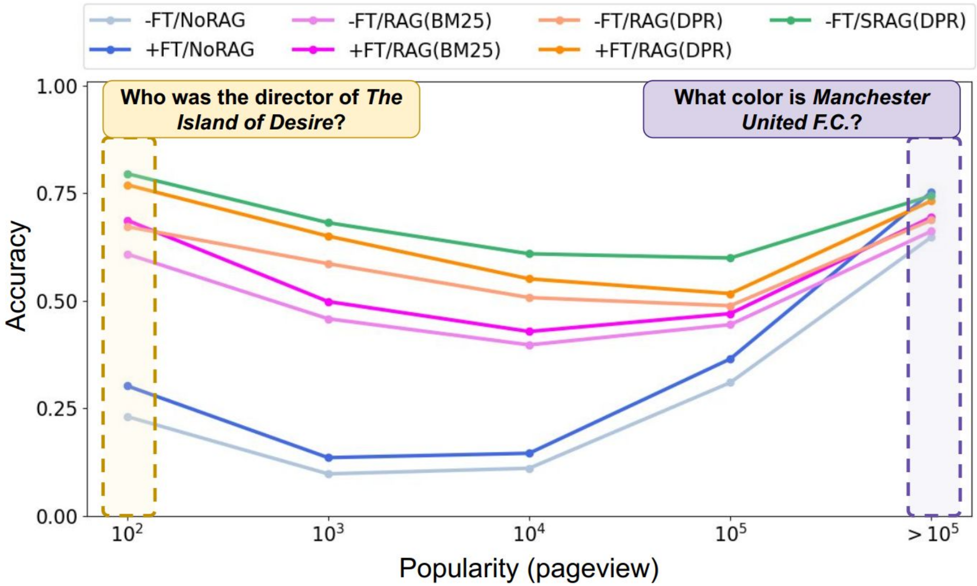
A clear correlation between the performance of the retriever and the overall QA accuracy

Results: Popularity

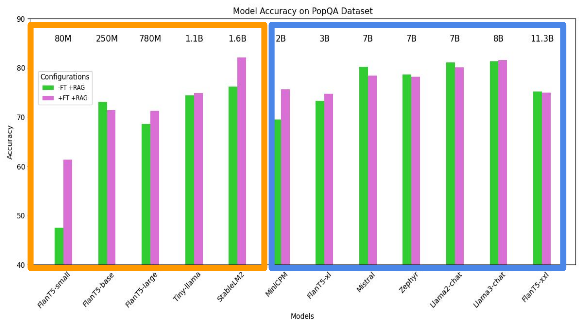


Results: Popularity





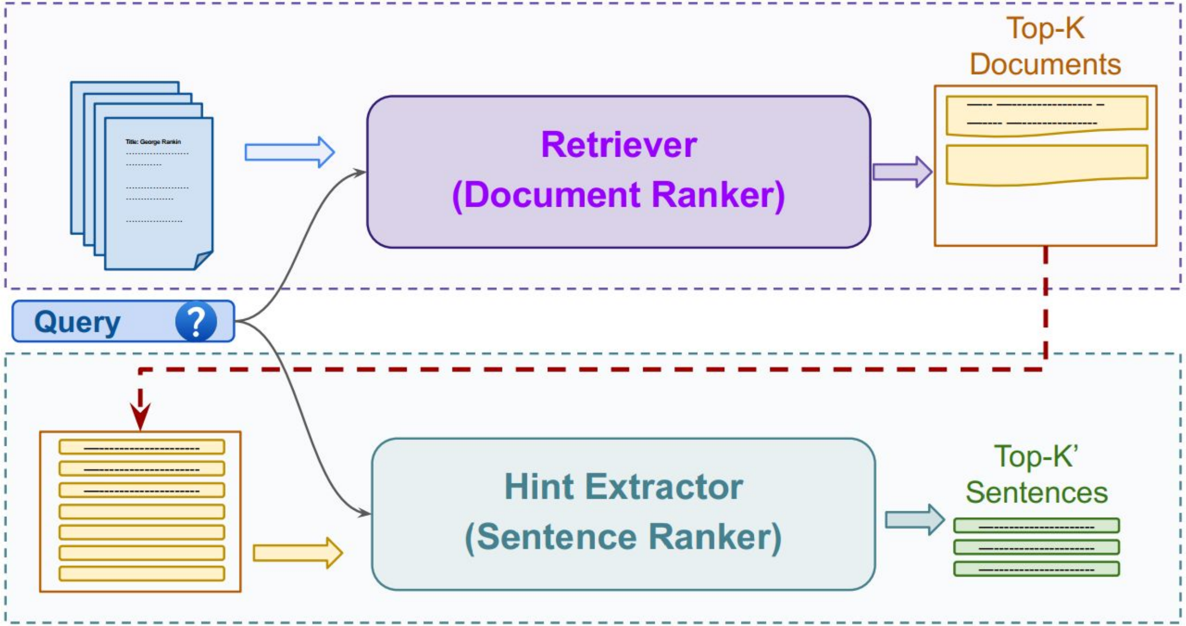
Research Questions



Fine-tuning demands a considerable amount of effort and resources

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?

Stimulus RAG



"Context: <hint><context> Question: <question>".

Results: Stimulus RAG

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

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

github.com/informagi

Questions?