Fine tuning is dead. Long live fine tuning?

The siren’s song of LLMs

Who am I

- Data Scientist at Zipcar
  - Trained models for demand prediction
- Led an ML education program about applied ML
  - Mentored 100+ fellows training models in NLP/CV
- Wrote a practical guide to building ML applications
  - Discusses model training in depth
- Staff ML Engineer at Stripe
  - Trained and improved fraud models, built some of the earliest MLOps
- Research Engineer at Anthropic
  - Finetuned Claude models, now working on interpretability

This talk = my opinion

I’ve been training models for 10 years

I don’t recommend it

1. Trends
2. Performance
3. Difficulty

Trends

“Be afraid of anything that sounds cool”

2009

~~Train models~~

What you want to do

Write SQL queries

What you **should** do

2012

~~Use deep learning~~

What you want to do

Use XGBoost

What you **should** do

2015

~~Invent a new loss function~~

What you want to do

Clean your data

What you **should** do

2023

~~Train your own LLM~~

What you want to do

Make better prompts

What you **should** do

2024

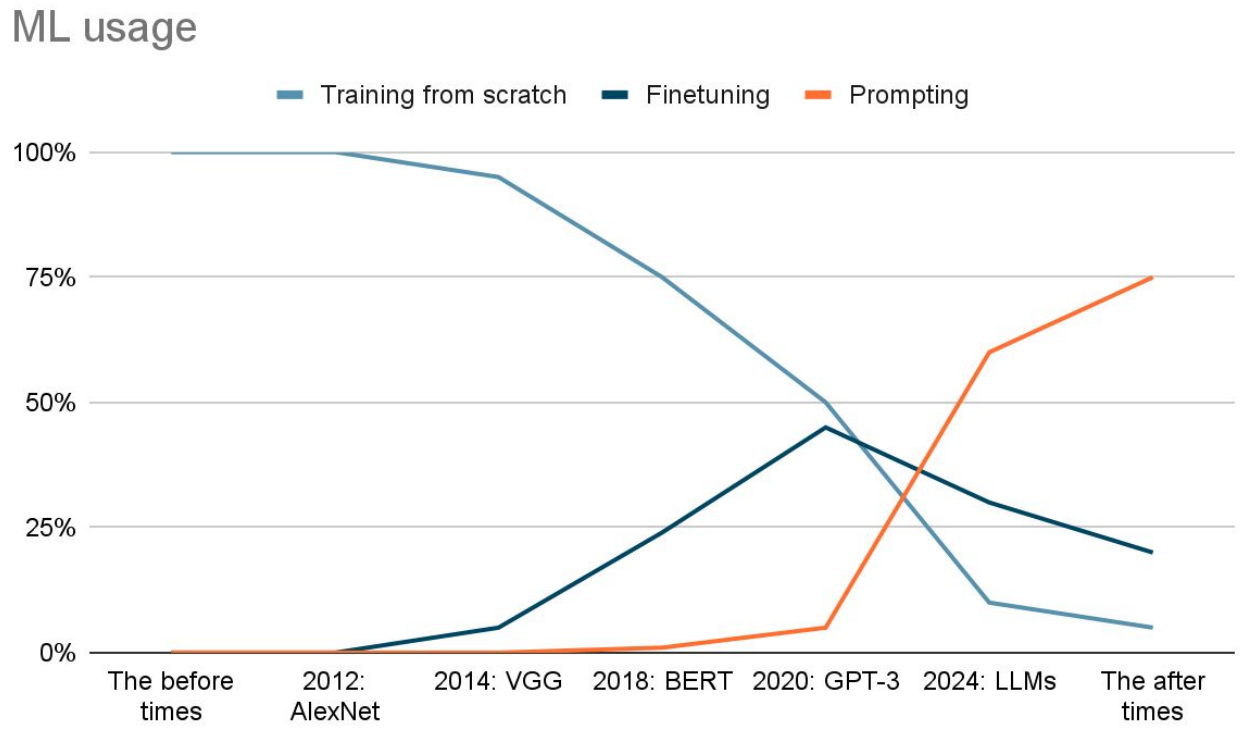
~~Finetune a LLM~~

What you want to do

Make better prompts

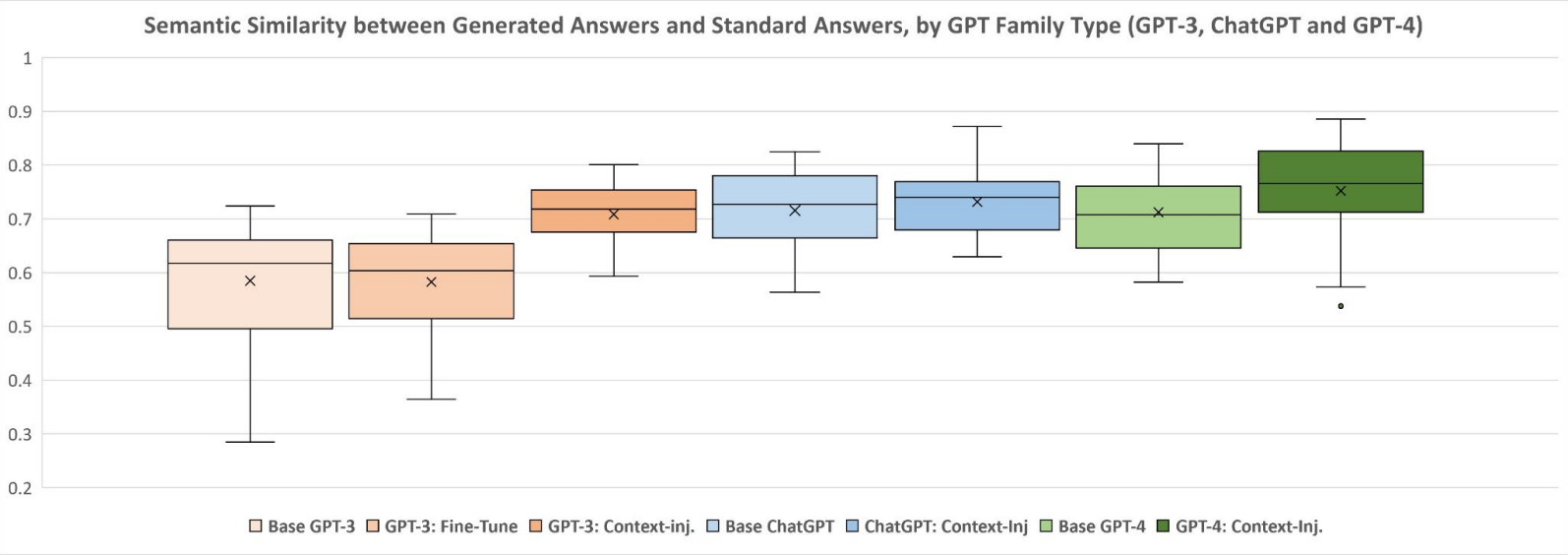
What you **should** do

The most productive way to leverage machine learning is changing



Performance

For most cases, finetuning does worse than prompts/RAG



Fine-tuning vs Context-Injection (RAG)

As models get bigger, the tradeoff gets worse

- Most of the gains come from RAG
- The impact of finetuning disappears for large models

FT	QA	-FT/ -RAG	-FT/ +RAG	+FT/ -RAG	+FT/ +RAG
FlanT5-small					
PEFT	E2E	3.05	26.13	5.53	22.91
	Prompt			7.01	49.85
Full	E2E			6.35	10.21
	Prompt			**8.52**	**49.88**
FlanT5-base					
PEFT	E2E	6.72	63.13	6.94	51.61
	Prompt			9.92	**63.29**
Full	E2E			8.63	24.17
	Prompt			**11.41**	60.26
FlanT5-large					
PEFT	E2E	8.41	58.12	8.22	55.26
	Prompt			12.15	**61.71**
Full	E2E			**16.23**	13.31
	Prompt			13.91	58.60

Table 2: Accuracy of base and fine-tuned models, both with and without RAG. The RAG results presented are based on ideal retrieval.

Fine Tuning vs. Retrieval Augmented Generation for Less Popular Knowledge

Although it isn't a clear win even for small models

	Base model	Base model + RAG	FT-reg	FT-par	FT-reg + RAG	FT-par + RAG
Mistral 7B	0.481	**0.875**	0.504	0.588	0.810	0.830
Llama2 7B	0.353	**0.585**	0.219	0.392	0.326	0.520
Orca2 7B	0.456	**0.876**	0.511	0.566	0.820	0.826

Fine-Tuning or Retrieval? Comparing Knowledge Injection in LLMs

Finetuning is not the solution for domain knowledge

Model	Fine-tuned	Fully correct (%)	+RAG
Llama-2-chat 13B		32%	49%
Vicuna		28%	56%
GPT-4		36%	60%
Llama2 13B	✓	29%	49%
GPT-4	✓	45%	61%

Table 20: Percent of answers that were fully correct, for base and fine-tuned models with and without RAG.

RAG vs Fine-tuning: Pipelines, Tradeoffs, and a Case Study on Agriculture

You are aiming at a moving target

How it started

Finance-Specific	BloombergGPT	GPT-NeoX	OPT-66B	BLOOM-176B
Financial Tasks	**62.51**	51.90	53.01	54.35
Bloomberg Tasks (Sentiment Analysis)	**62.47**	29.23	35.76	33.39

General-Purpose	BloombergGPT	GPT-NeoX	OPT-66B	BLOOM-176B	GPT-3
MMLU	**39.18**	35.95	35.99	39.13	**43.9**
Reading Comprehension	**61.22**	42.81	50.21	49.37	**67.0**
Linguistic Scenarios	**60.63**	57.18	58.59	58.26	**63.4**

BloombergGPT: A Large Language Model for Finance

You are aiming at a moving target

How it's going

Model	FinQA	ConvFinQA
ChatGPT(0)	48.56	59.86
ChatGPT(3)	51.22	/
ChatGPT(CoT)	63.87	/
GPT-4(0)	68.79	**76.48**
GPT-4(3)	69.68	/
GPT-4(CoT)	**78.03**	/
BloombergGPT(0)	/	43.41
GPT-NeoX(0)	/	30.06
OPT66B(0)	/	27.88
BLOOM176B(0)	/	36.31
FinQANet(fine-tune)	68.90	61.24
Human Expert	91.16	89.44
General Crowd	50.68	46.90

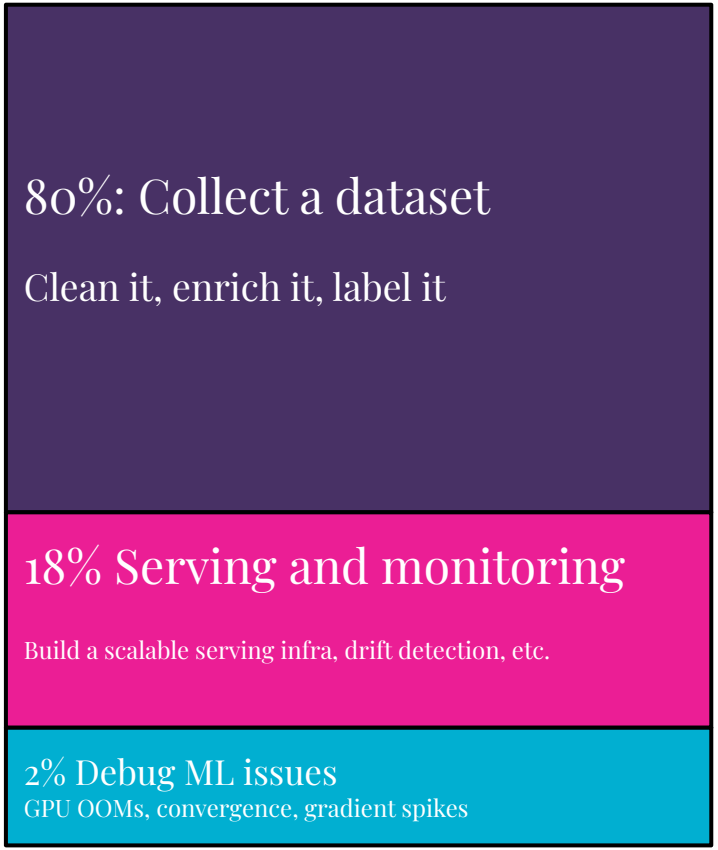
Are ChatGPT and GPT-4 General-Purpose Solvers for Financial Text Analytics? A Study on Several Typical Tasks

Difficulty

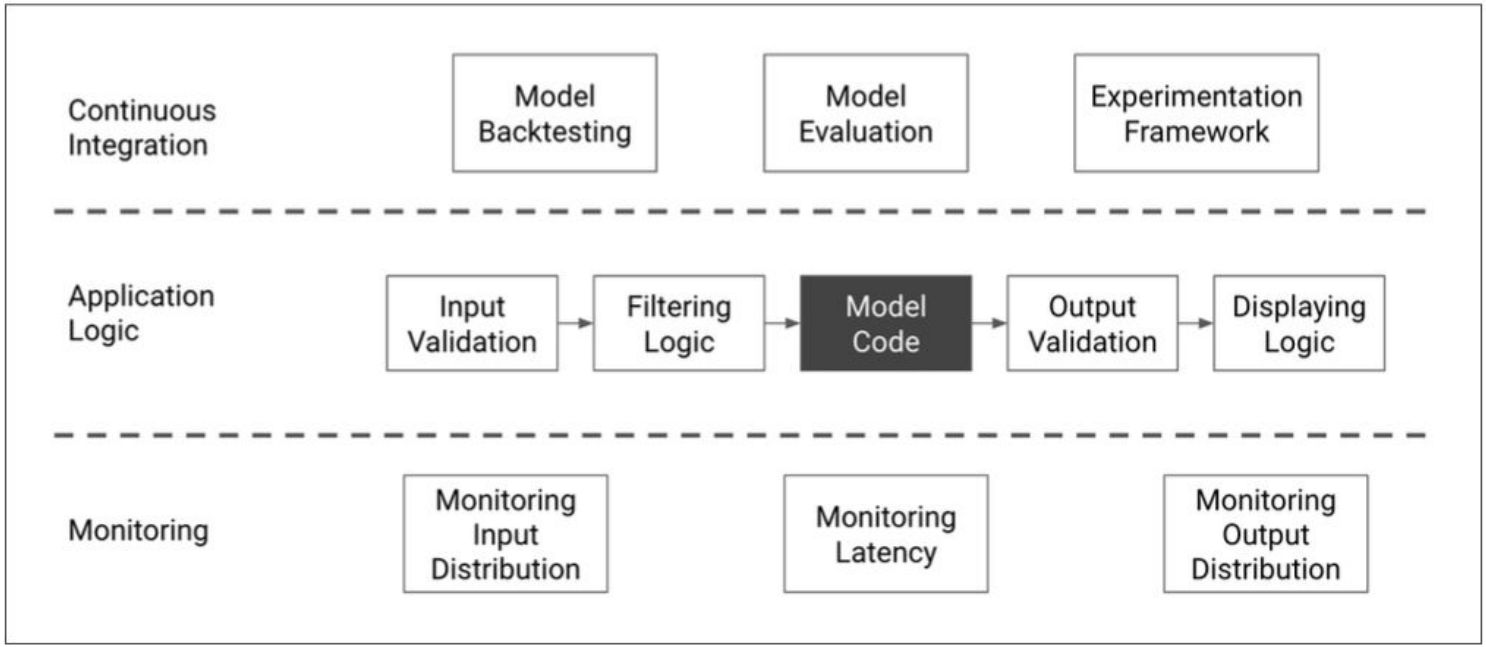
Optimal machine learning: average time spent per task

Not pictured because it doesn't/shouldn't happen:

- Cool architecture research



Machine learning is hard **even if you don't train the models!**

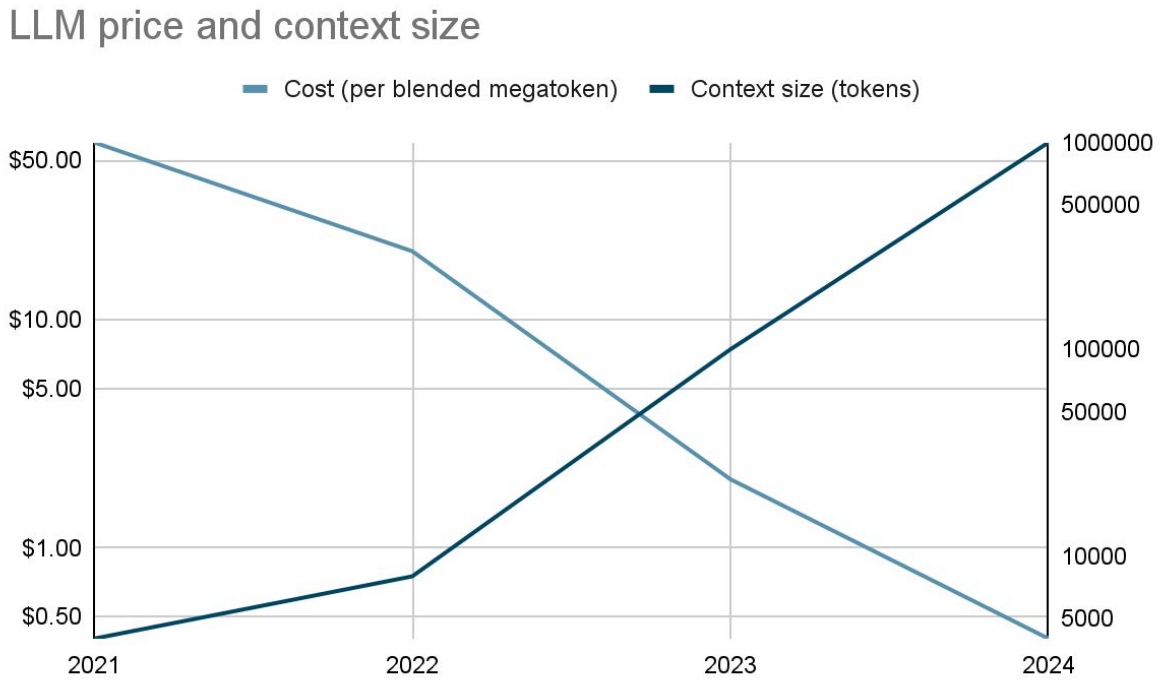


Building ML Powered Applications

You have better things to do!

- Building an evaluation set that is:
  - representative, large, and easy to run
- Spending days working on prompts and RAG
- Investing in monitoring and error detection

Skate to where the puck is going



A surprisingly balanced conclusion

Keep in mind that:

- Finetuning is:
  - expensive and complex
  - has become less valuable
  - often underperforms simpler approaches
- Models are continuously becoming:
  - cheaper
  - smarter
  - faster
  - longer context

So:

- Always start with:
  - prompting
  - making a train/test set
  - rag
- Treat finetuning as a niche/last resort solution
  - like cloud vs on prem