



Small Data SF

Workshop: Going from Single Model to Compound AI Systems

Mikiko Bazeley
Ray Thai



What is Fireworks?



What is Fireworks?

Fireworks AI is an Generative AI inference platform focused on:

- Performance
- Production-readiness
- Customizability

Fireworks AI is best-known for providing highly-optimized models serving across modalities:

- Text
- Image (official backend for Stability AI)
- Embeddings



Who uses Fireworks?



Who is Fireworks for?

- Developers and engineers building advanced AI applications.
- Data scientists and researchers working with LLMs, multimodal models, and generative AI.
- Enterprises deploying scalable AI systems for production.
- Hackathon participants, startups, and tech teams leveraging Fireworks for innovation.





Upwork HubSpot

Quora

DOORDASH

verizon[✓]

Cursor



Sourcegraph

CRESTA

liner

tome

sizzle

wrtn.

stability.ai

Why use Fireworks?



Why use Fireworks?

- Simplifies the process of connecting to, deploying, and fine-tuning models.
- Offers scalability from development to production, with a focus on compound AI systems.
- Enables integration with other key AI infrastructure components (vector stores, APIs).
- High performance with infrastructure support for RAG, function-calling, and fine-tuning.



How?



Fireworks Compound AI System

Fireworks Compound AI SDK

Expanded knowledge

Fireworks Adaptation Engine

Grounding

Fireworks Distributed Inference Engine

Interactive latency

High cost efficiency

Why use Fireworks?

- Simplifies the process of connecting to, deploying, and fine-tuning models.
- Offers scalability from development to production, with a focus on compound AI systems.
- Enables integration with other key AI infrastructure components (vector stores, APIs).
- High performance with infrastructure support for RAG, function-calling, and fine-tuning.



Fireworks Compound AI System

Fireworks Compound AI SDK

Expanded knowledge

Text



Audio



Embedding



Image



Vision-language



Fireworks Adaptation Engine

Grounding

Fireworks Distributed Inference Engine

Interactive latency

High cost efficiency

100+ Top Quality Models

Text

Image

Audio

Vision

Embedding

Generation



Fireworks Compound AI System

Fireworks Compound AI SDK

Expanded knowledge

Fireworks Adaptation Engine

Grounding

Fireworks Distributed Inference Engine

Interactive latency

Decentralized
learning

Semantic Caching

On-demand

BYOC



Google Cloud

ORACLE®

Fireworks Compound AI System

Fireworks Compound AI SDK

Expanded knowledge

Fireworks Adaptation Engine

Grounding

Fireworks Distributed Inference Engine

FireAttention

Disaggregated
Serving

Semantic Caching

Multi-region

Multi-cloud

On-demand

BYOC



NVIDIA



ORACLE®

Fireworks Distributed Inference Engine

Semantic Caching

Middleware
Prompt cache
KV cache

Disaggregated Serving

Hybrid sharding for multiple modalities
Speculative runtime
Efficient batching

FireAttention (GenAI-specific GPU kernels)

MQA, GQA, MLA, Distributed, long context, quality-preserving quantization



On-demand

BYOC

Multi-region

Multi-cloud



Why use Fireworks?

- Simplifies the process of connecting to, deploying, and fine-tuning models.
- Offers scalability from development to production, with a focus on compound AI systems.
- Enables integration with other key AI infrastructure components (vector stores, APIs).
- High performance with infrastructure support for RAG, function-calling, and fine-tuning.



Fireworks Compound AI System

Fireworks Compound AI SDK

Expanded knowledge

Text



Audio

Embedding



Image



Vision-language



Fireworks Adaptation Engine

Grounding

Fireworks Distributed Inference Engine

FireAttention

Disaggregated Serving

Semantic Caching

Multi-region

Multi-cloud

On-demand

BYOC



NVIDIA



Google Cloud

ORACLE

Fireworks Compound AI System

Fireworks Compound AI SDK

Expanded knowledge

Fireworks Model Stock

Text



Audio



Embedding



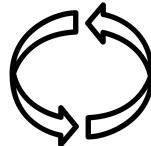
Image



Vision-language



Fireworks Adaptation Engine



Supervised
Fine-tuning

Self-tune

Personalization

Fireworks Distributed Inference Engine

FireAttention

Disaggregated
Serving

Semantic Caching

Multi-region

Multi-cloud

On-demand

BYOC



NVIDIA



Google Cloud

ORACLE

Why use Fireworks?

- Simplifies the process of connecting to, deploying, and fine-tuning models.
- Offers scalability from development to production, with a focus on compound AI systems.
- Enables integration with other key AI infrastructure components (vector stores, APIs).
- **High performance with infrastructure support for RAG, function-calling, and fine-tuning.**



Fireworks Compound AI System

Fireworks Compound AI SDK

FireFunction

JSON/Grammar mode

...

Fireworks Model Stock

Text

Audio

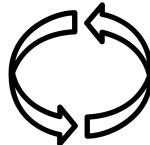
Embedding

Image

Vision-language



Fireworks Adaptation Engine



Supervised
Fine-tuning

Self-tune

Personalization

Fireworks Distributed Inference Engine

FireAttention

Disaggregated
Serving

Semantic Caching

Multi-region

Multi-cloud

On-demand

BYOC



NVIDIA



Why use Fireworks?

- Simplifies the process of connecting to, deploying, and fine-tuning models.
- Offers scalability from development to production, with a focus on compound AI systems.
- **Enables integration with other key AI infrastructure components (vector stores, APIs).**
- High performance with infrastructure support for RAG, function-calling, and fine-tuning.



Go from
experimentation to
production quickly



Production-ready options from idea to enterprise-scale

Serverless Inference

- Pay by token
- Constant \$ per token
- No hardware (HW) configuration options

On-demand

- Pay by GPU-second
- Choose HW set-up & software modifications

Enterprise Reserved

- Pay by GPU reservation w/ bulk discount
- Personalized HW set-up + SLAs



Better with less scale

Better with more scale



Who is Fireworks?



Who is Fireworks?



WHO WE ARE

Fastest and most efficient inference engine to build production-ready, compound AI systems.

FOUNDERS

Founding Team Led PyTorch at Meta & GCP

- Serving 50 TRN+ inferences per day



FOUNDED

2022, Series-B

CUSTOMERS

Uber DOORDASH HubSpot verizon[✓]

INVESTORS

SEQUOIA

BENCHMARK

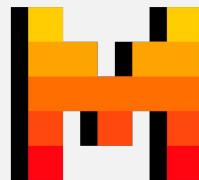


Fireworks - productionizing open-source LLMs



Fireworks Team

PyTorch & Google AI



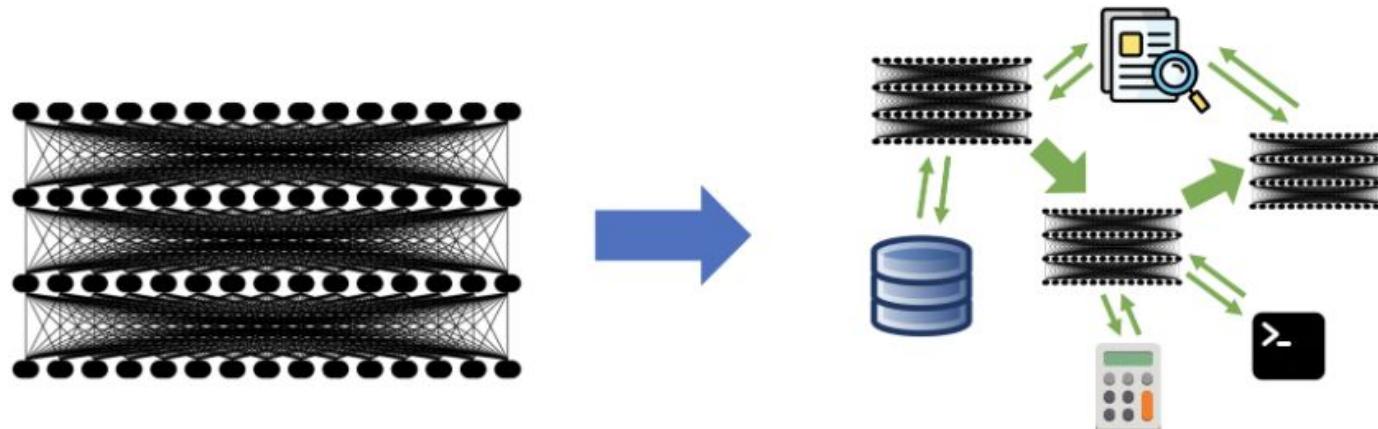
Quora



Compound AI is what's coming up



Compound AI System



Increasingly many new AI results are from compound systems



Multi-mod{e/a}lity is Critical in Production



Sales Call Report Example

Contact: Sue Smith
Title: Director of Operations
Company: Widgets, Inc.
Phone Number: (555) 555-5555
Call Purpose: Schedule demo
Call Date: 06/07/2021
Call Time: 2:00 pm
Call Duration: 17 minutes

Prep Notes:
Widgets, Inc. has between 20-50 employees, making it a good fit for our solution. Spoke to Bob last week and earned a call with Sue, the decision maker. Bob mentioned that she's currently bullish on their current solution.

Call Plan:
Position ourselves as a more tailored solution than their current one and schedule live demo to demonstrate fit.

Call Summary and Results:
Due to Sue's pain points with managing and processing, she's open to a live demo, but she made it clear that our pricing model may be a deal-breaker due to their high volume.

Follow-Up Date: 06/21/2021

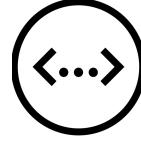
Text



Audio



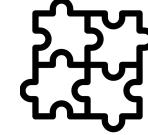
Image



Embedding



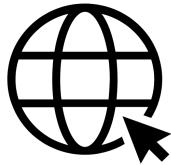
Vision



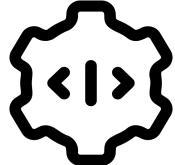
Multimodal



API Access is Critical in Production



Internet
Search



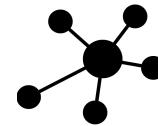
Retrieval
APIs



Database



Documents



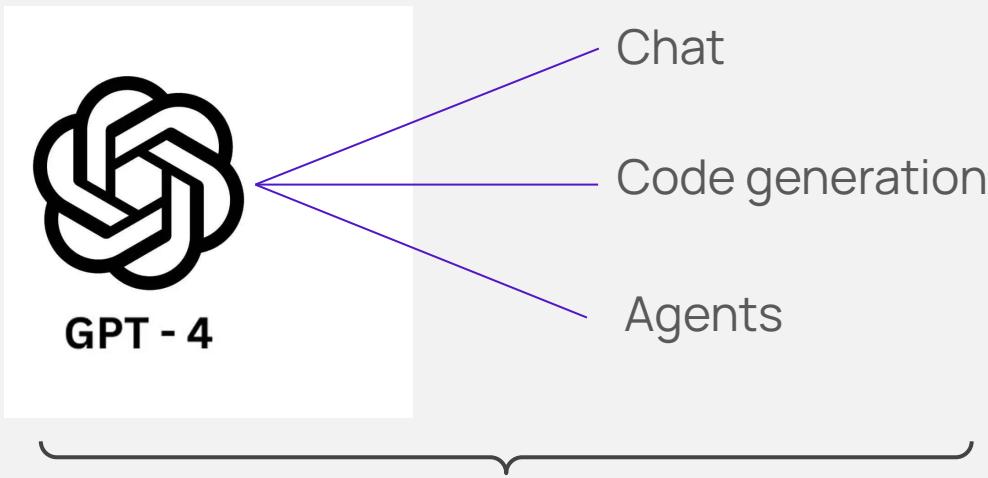
Graph



The role alternative model providers play in the ecosystem



Closed source - Huge model served identically

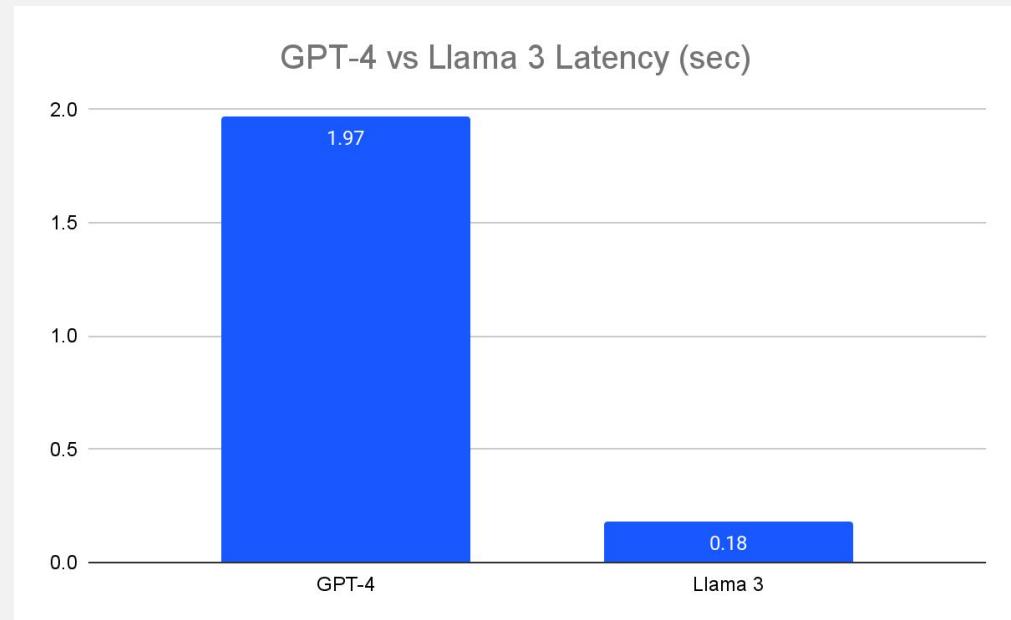


High costs and poor UX in production

High Latency

Slow processing times
hinder real-time
performance

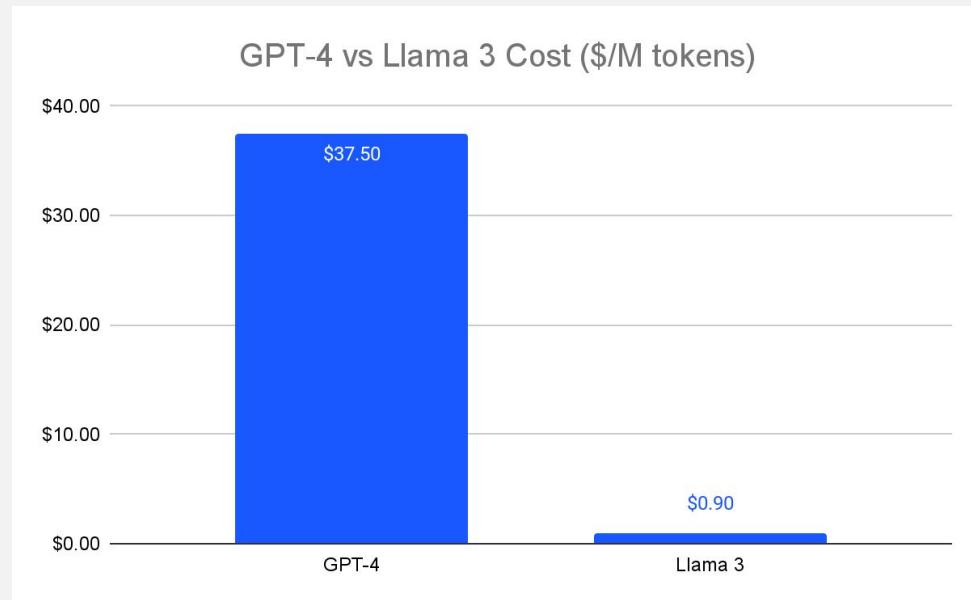
Impact: Low user
engagement & product
adoption



Elevated Costs

High operational costs -
1B tokens = \$37k more
expensive daily, \$13M
yearly

Impact: Unsustainable
business model



Limited Domain Adaptability

General-purpose models typically lack knowledge in specialized areas

Impact: Generic & irrelevant responses, no moat

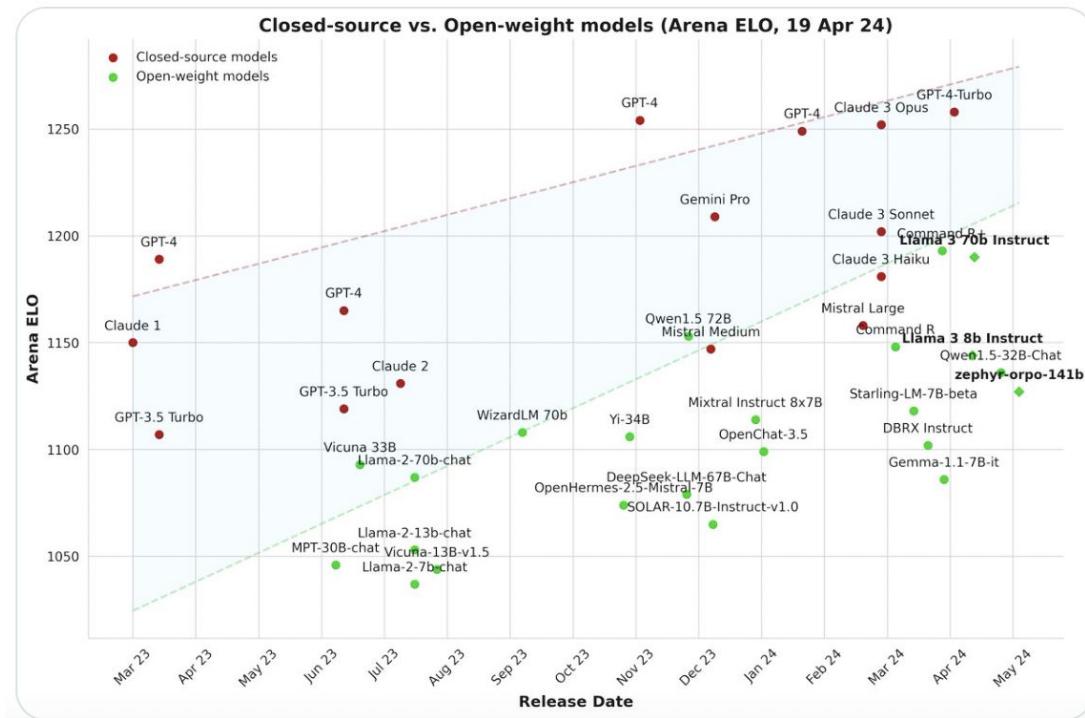
50k+

Open-source GenAI models

\$0

Cost to serve custom or tuned models

Quality Convergence: Open vs Closed Source Models



Open-source unlocks real business at scale



Chat



Code generation



Agents

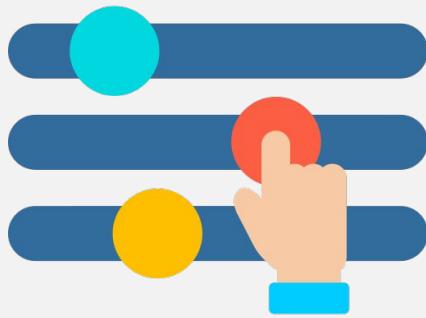


Data Processing

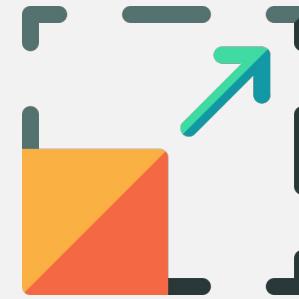
Using open-source models in production is still challenging



Complex Set-up



Not Optimized



Not Production-Ready

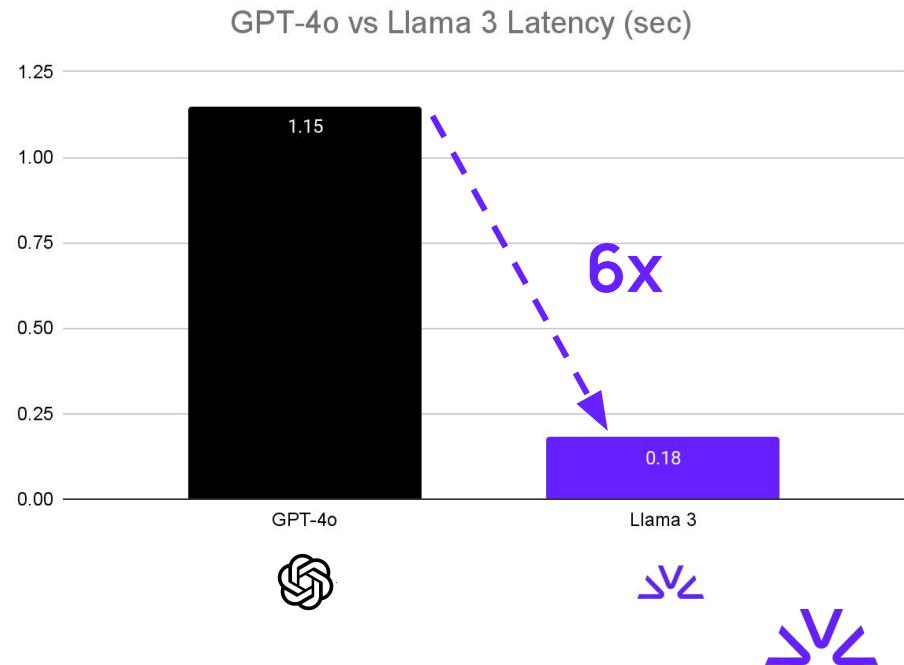
Also, how fast?



6x improvement on Latency vs. GPT-4o

Slow processing times
hinder product
engagement and adoption

Our impact: Viable product
experience

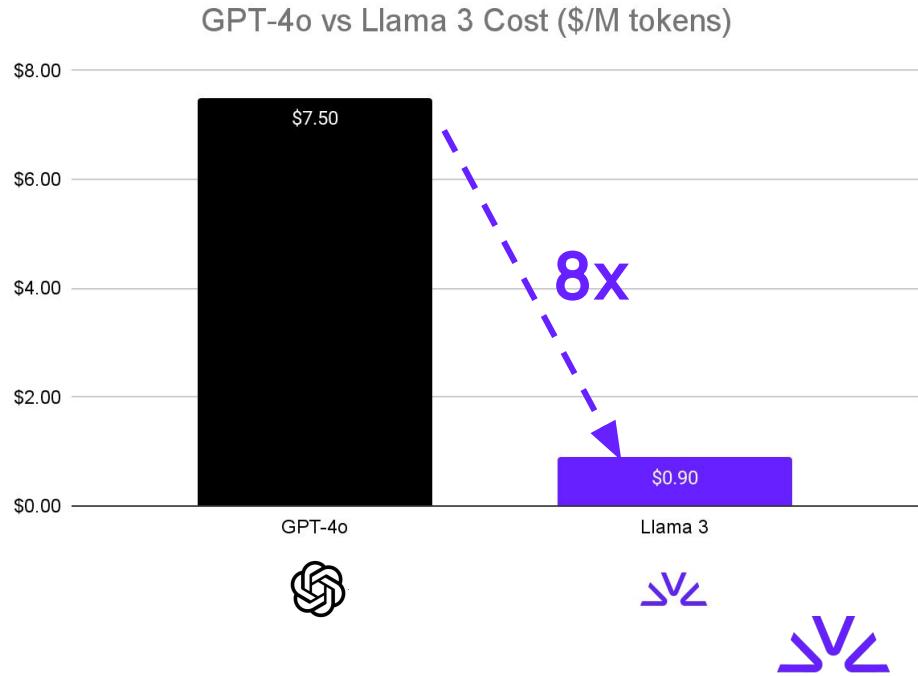


Source: Berkeley Function-Calling Leaderboard

8x reduction in costs vs. GPT-4o

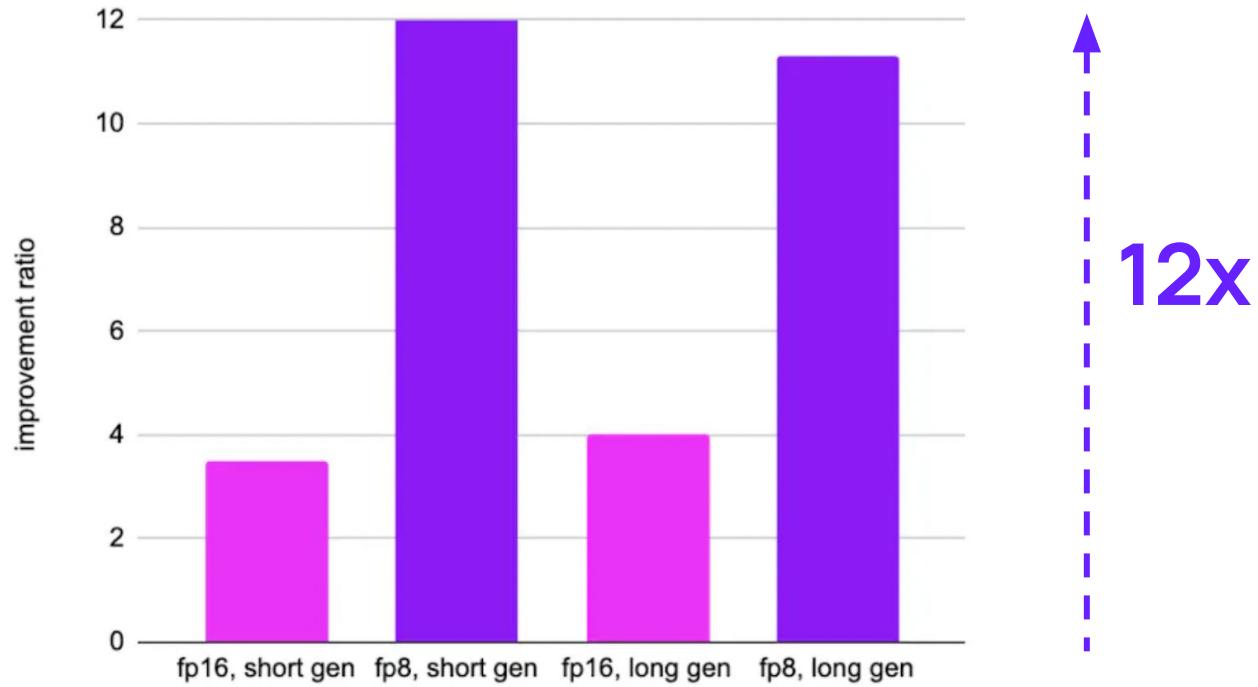
High operational costs -
1B tokens = \$6.5k more
expensive daily, \$2.4M
yearly

Our impact: Sustainable
business model



Costs calculated with a 3:1 input:output prompt ratio

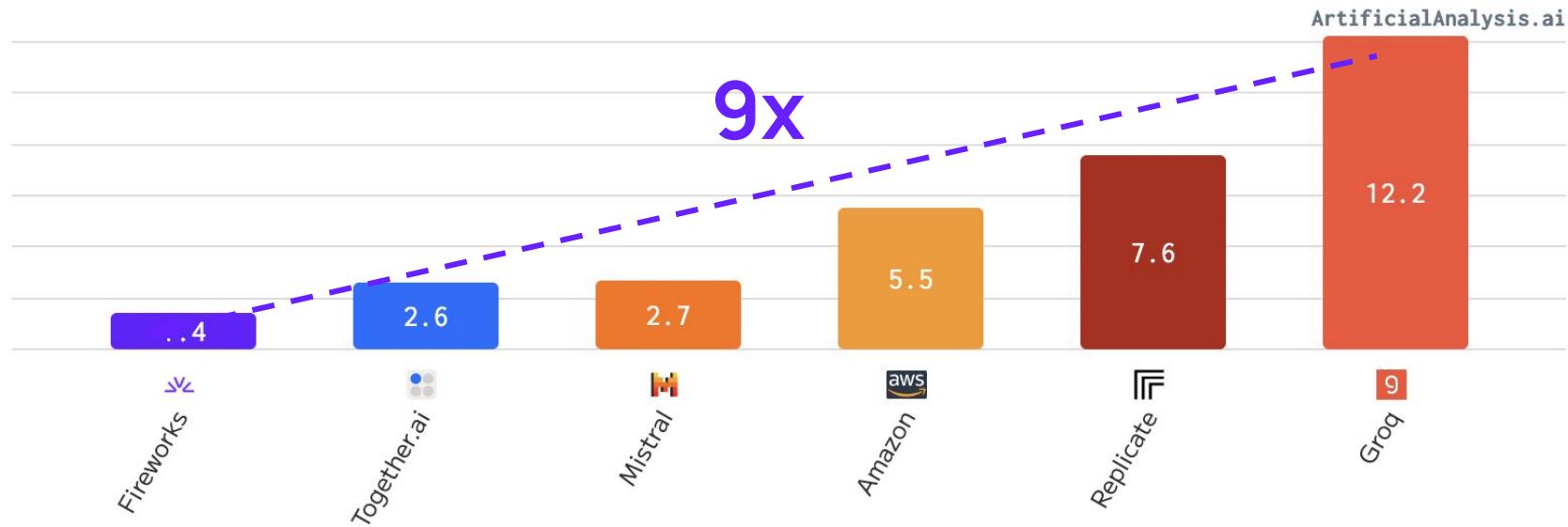
Long Context Prompt Latency Improvement Over vLLM



LLM with RAG

Total Response Time

Seconds to Output 100 Tokens; Lower is better

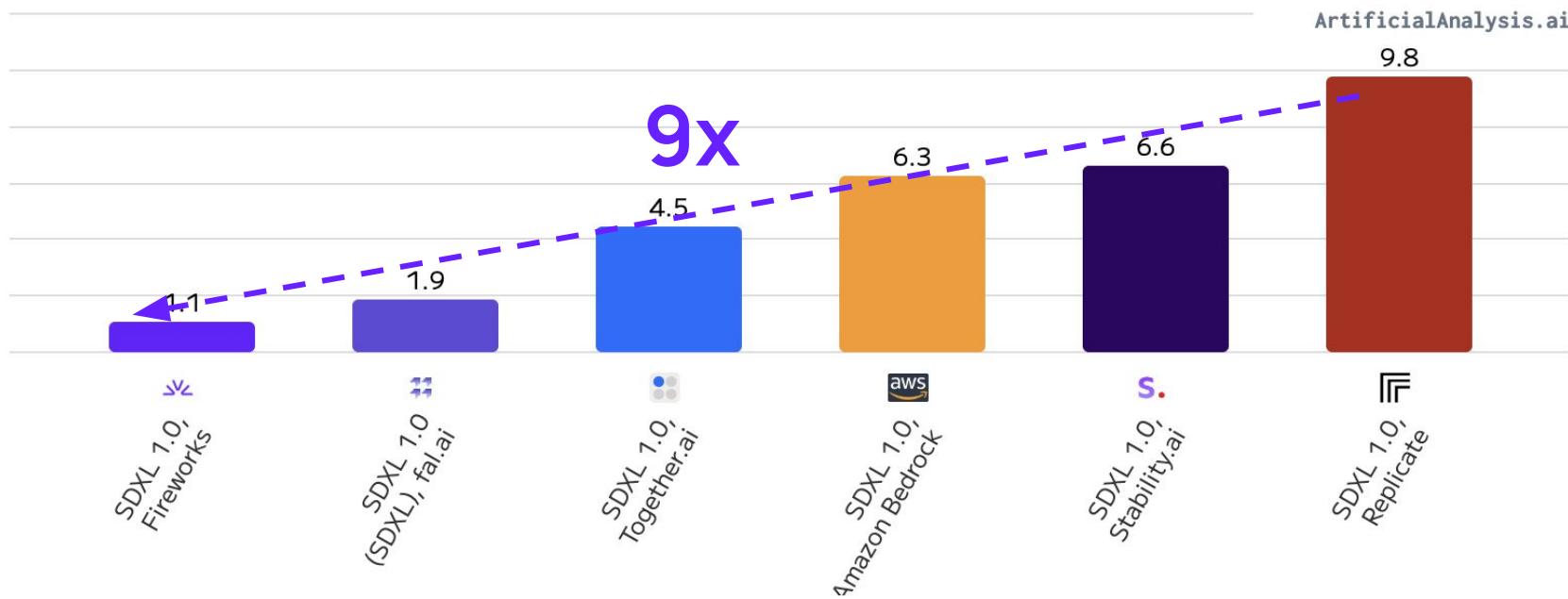


Independent benchmark from ArtificialAnalysis.ai, Mixtral 8x7b Instruct, 10 parallel queries, 10k token prompt

Image Generation Speed

Generation Time

Generation time: Seconds to generate 1 image, Lower is better



Independent benchmark from ArtificialAnalysis.ai, SDXL 1.0 Image Generation

Patterns of developing compound AI systems



Workshop Overview Slide

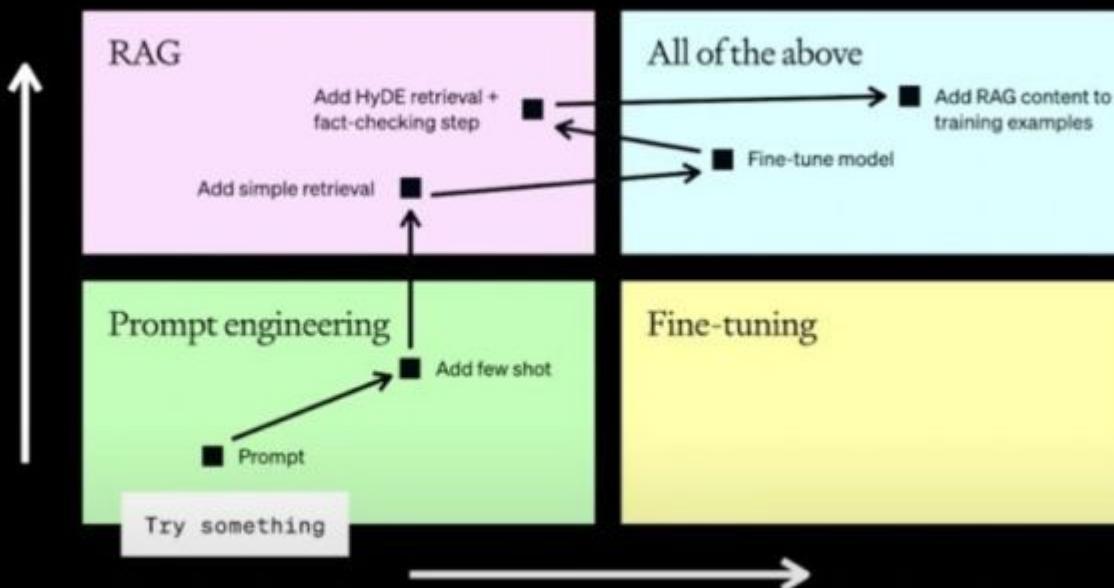
- Connecting to models using Fireworks and building a Streamlit app for model comparison.
- Deploying LLMs for RAG and function-calling workflows.
- Fine-tuning LLMs and scaling compound AI systems.
- Post-training practices: LoRAs, evaluation, etc.



The optimization flow

Context
optimization

What the model
needs to know



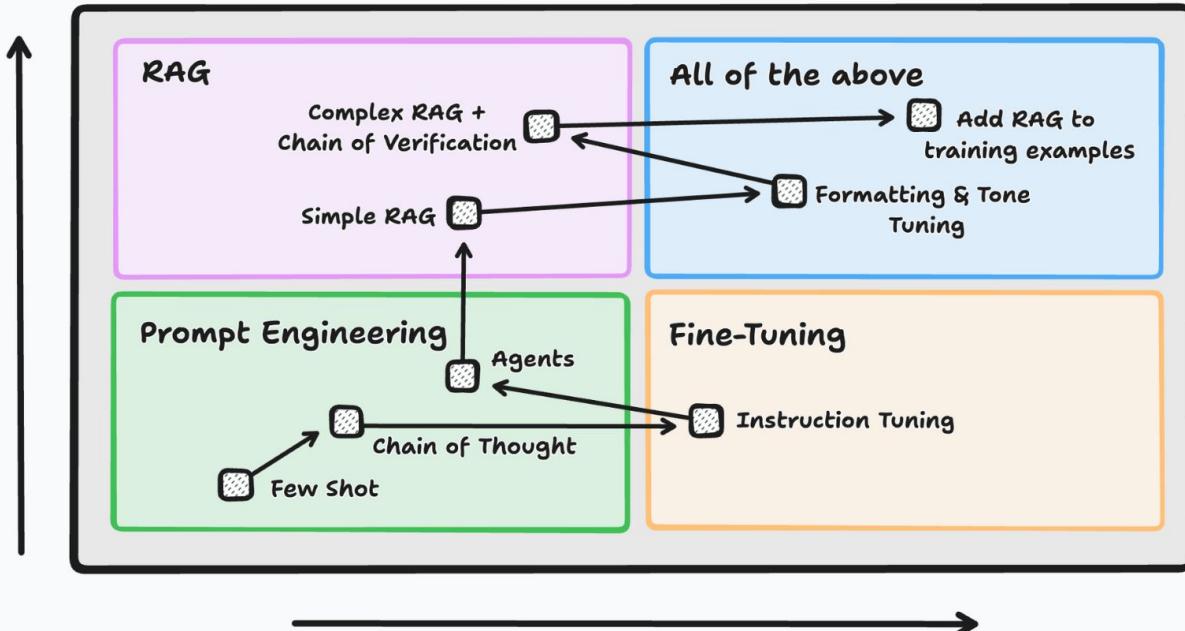
LLM optimization

How the model needs to act



Context Optimization

What is told
to the model



LLM Optimization

How the model acts



Repo

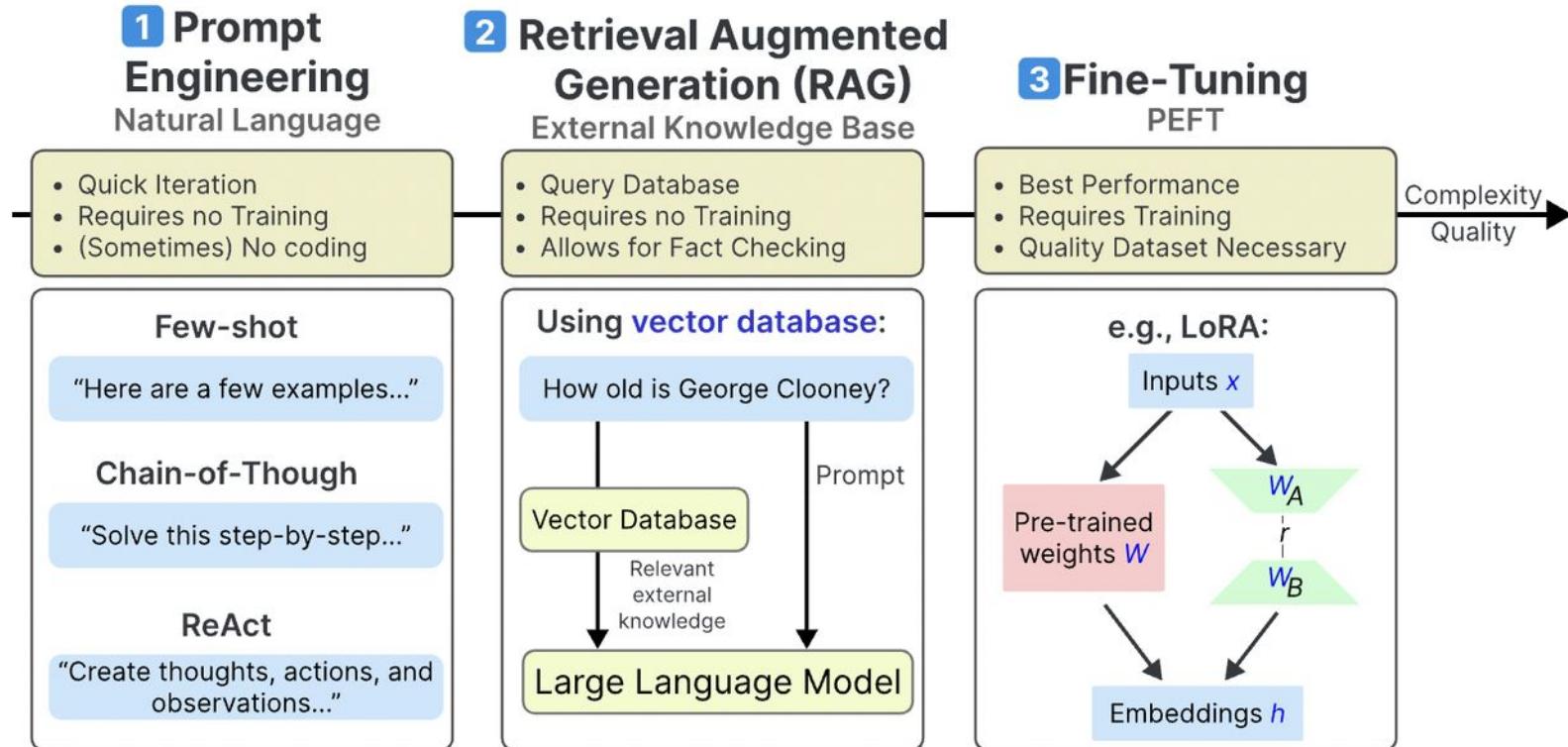
- Part 0: Using LLMs 101
- Part 1: Connecting to Fireworks & Building a Model Comparison App
- Part 2: Deploying an LLM with RAG
- Part 3: Deploying a Function-Calling LLM as part of an Agentic Workflow
- Part 4: Fine-tuning Base and Function-Calling LLMs
- Part 5: Developing a Minimum Viable Compound AI System
- Part 6: The "Ops" Side of Compound AI Systems



Workshop Overview Slide

- Connecting to models using Fireworks and building a Streamlit app for model comparison.
- Deploying LLMs for RAG and function-calling workflows.
- Fine-tuning LLMs and scaling compound AI systems.
- Post-training practices: LoRAs, evaluation, etc.





Let's get setup



The steps

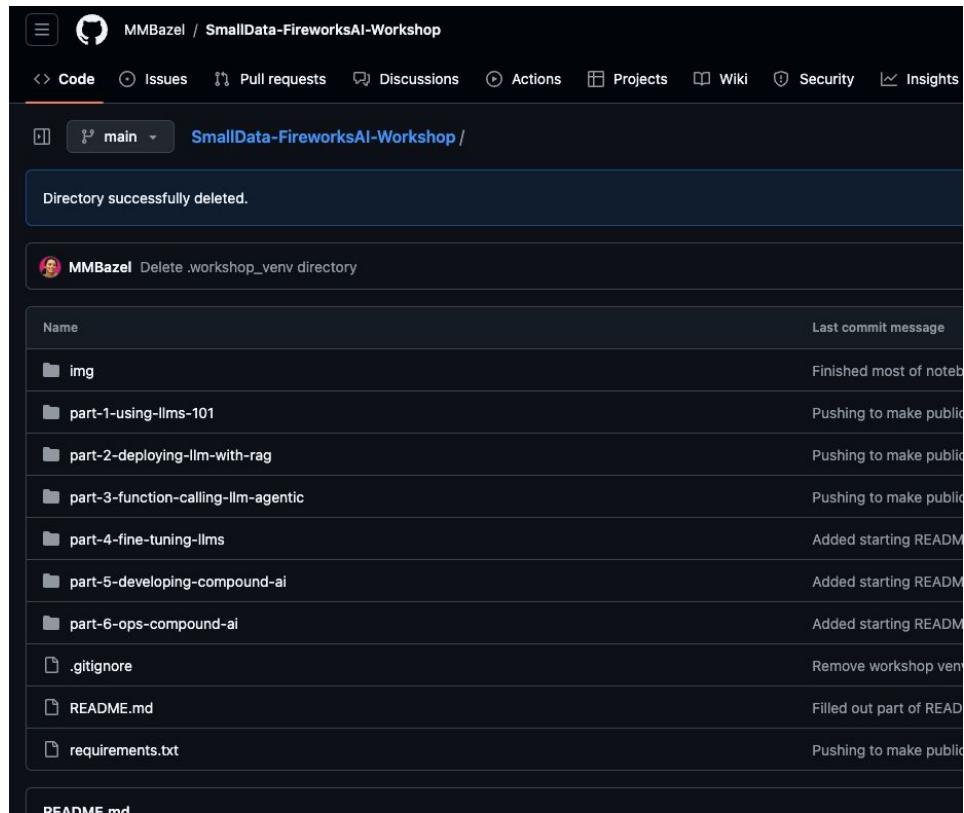
- The Github repo: Contains sample code we'll walk through, including notebooks
 - URL: <https://bit.ly/SmallDataSF-FW>
- Sign up for an account on Fireworks: <https://fireworks.ai/>
- Join the Discord - the #smalldata-workshop channel:
<https://discord.gg/KKMTQNdf>
 - Fill out form here: <https://forms.gle/uVmbPiSWTVFiqLEu7>
Once you've signed up for your account on Fireworks,
copy-paste your account ID in the thread so I can give \$20
credit



The steps

- The Github repo:
Contains sample
code we'll walk
through, including
notebooks
 - URL:

<https://bit.ly/SmallDataSF-FW>



Pattern 1: Using an LLM 101



Section overview

What we'll cover in this section:

- Introduction to LLMs and their capabilities.
- Setting up the Fireworks connection.
- Basics of querying models.

Plus: Connecting to Fireworks & Building a Model Comparison App
Slide

- Steps to build a Streamlit app for comparing models hosted on Fireworks.
- Link to guide on querying text models.
- Link to guide on querying embedding models.



Github

The screenshot shows a GitHub repository for "SmallData-FireworksAI-Workshop". The repository has a single commit from "Mikiko Bazeley" pushing files to the "main" branch. The commit messages are:

- "Pushing to make public" (for README.md)
- "Finished most of notebook for Part 1" (for app.py)

Below the repository view, there is a workshop page with the following content:

Part 1: Connecting to Fireworks & Building a Model Com

Objectives:

In this part of the workshop, you will:

- Connect to Fireworks AI's platform to access a list of hosted models.
- Utilize a Streamlit app that allows users to compare different models from Fireworks via a dropdown menu.
- Perform initial "vibe checks" by querying the selected models.

Relevant Concepts/Skills:

Please complete this section with



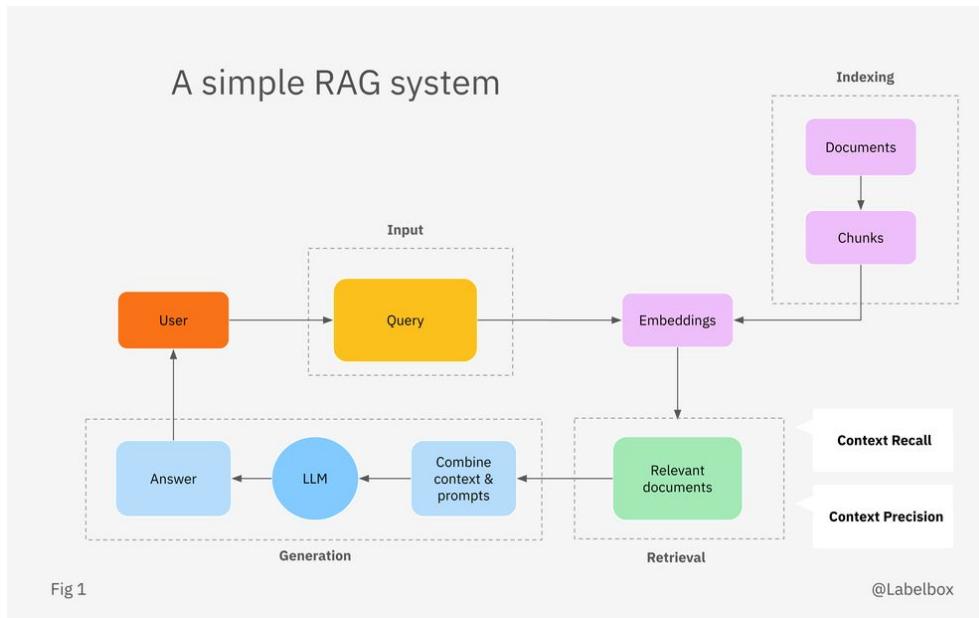
Pattern 2: A simple RAG



Section overview

What we'll cover in this section:

- Overview of Retrieval-Augmented Generation (RAG).
- Steps to deploy an LLM with a vector store for RAG.



Github

Code Issues Pull requests Discussions Actions Projects Wiki Security Insights

main SmallData-FireworksAI-Workshop / part-2-deploying-llm-with-rag /

Mikiko Bazeley and Mikiko Bazeley Pushing to make public

Name	Last commit message
..	
chromadb	Completed part 2 notebo
data_lol	Completed part 2 notebo
scraping_misc	Completed part 2 notebo
RAG-with-Fireworks.ipynb	Completed part 2 notebo
README.md	Pushing to make public

README.md

Part 2: Deploying an LLM with Retriever

Objectives:

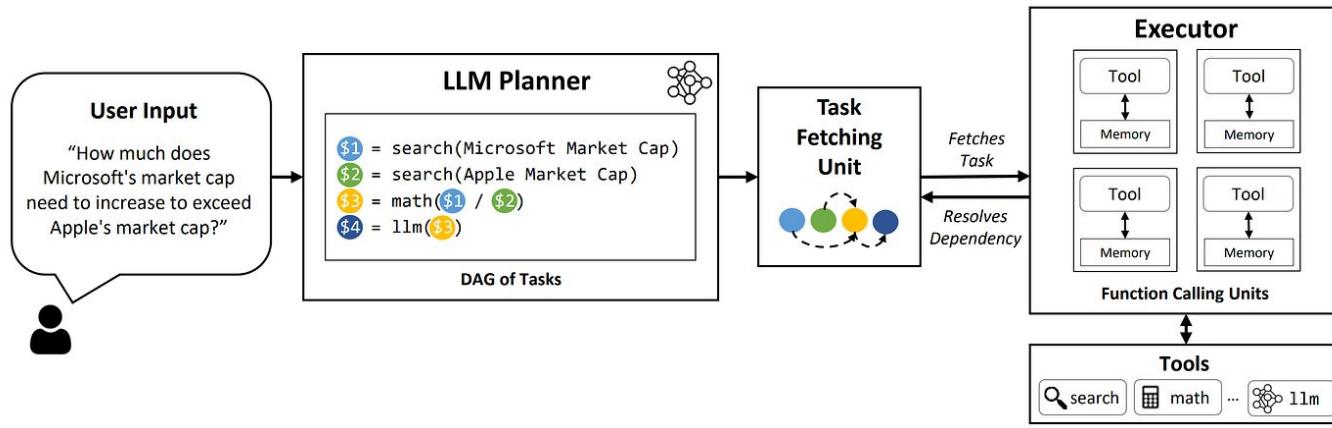


Pattern 3: Function calling and compound AI



Section overview

- Explanation of function-calling LLMs.
- Steps to implement an agentic workflow.
- Link to guide on using function-calling with Fireworks.



LLMs are great at interfacing with humans

Llama 3 70B Instruct

Meta developed and released the Meta Llama 3 family of large language models (LLMs), a collection of pretrained and instruction tuned generative text models in 8 and 70B sizes. The Llama 3 instruction tuned models are optimized for dialogue use cases and outperform many of the available open source chat models on common industry benchmarks.

\$0.9/M Tokens 8,192 Max Context

Chat Completion



Can you give me a really short overview on Nvidia?



Here's a brief overview of Nvidia:

Nvidia Corporation is an American technology company that specializes in designing and manufacturing graphics processing units (GPUs) and high-performance computing hardware. Founded in 1993 by Jensen Huang, Chris Malachowsky, and Curtis Priem, Nvidia is headquartered in Santa Clara, California.

Key products and areas of focus:

1. **GeForce GPUs**: Nvidia's flagship product line, providing graphics processing for gaming, professional visualization, and artificial intelligence.
2. **Datacenter and AI**: Nvidia's datacenter products, including Tesla V100 and T4 GPUs, power AI, deep learning, and cloud computing.
3. **Autonomous vehicles**: Nvidia's Drive platform provides AI computing for self-driving cars and trucks.
4. **Gaming**: Nvidia's GeForce GPUs and gaming platforms, such as GeForce Now, enable PC gaming and esports.

Notable achievements:

- Pioneer in GPU-accelerated computing
- Leader in AI computing and deep learning
- Partner with major companies like Google, Microsoft, and Amazon Web Services
- Acquired Mellanox Technologies in 2020 to expand its datacenter capabilities

That's a quick snapshot of Nvidia!

Type a message



LLMs are limited in interfacing with external tools

Llama 3 70B Instruct

Meta developed and released the Meta Llama 3 family of large language models (LLMs), a collection of pretrained and instruction tuned generative text models in 8 and 70B sizes. The Llama 3 instruction tuned models are optimized for dialogue use cases and outperform many of the available open source chat models on common industry benchmarks.

\$0.9/M Tokens 8,192 Max Context

Chat Completion

What's the stock price of Nvidia

As of my knowledge cutoff, the current stock price of NVIDIA Corporation (NVDA) is around \$529.00 USD per share. However, please note that stock prices can fluctuate rapidly and may be different when you check.

To get the most up-to-date and accurate information, I recommend checking a reliable financial website or platform, such as:

1. Yahoo Finance: www.finance.yahoo.com
2. Google Finance: www.google.com/finance
3. Bloomberg: www.bloomberg.com
4. Nasdaq: www.nasdaq.com

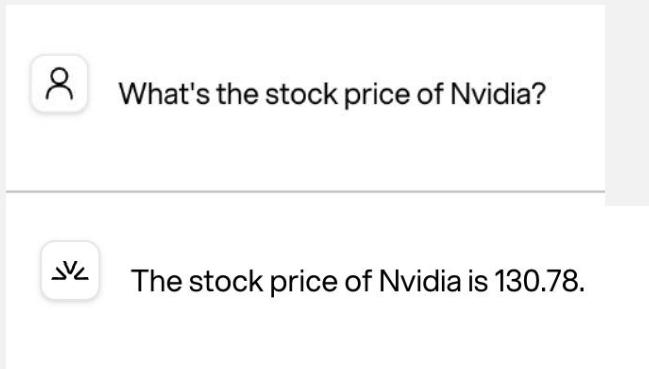
You can enter "NVDA" or "NVIDIA Corporation" in the search bar to get the current stock price and other relevant information.

302ms initial latency | 199.09 tokens/s



What is function calling?

- Functions give models the ability to interface with external information and tools



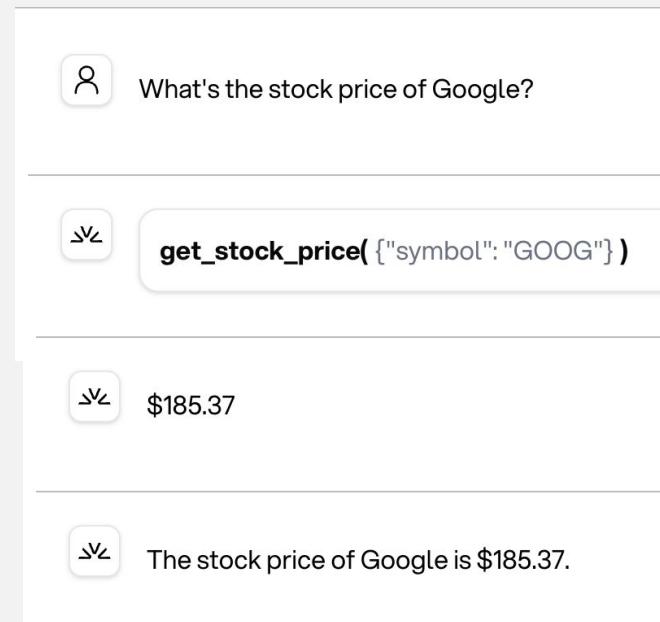
Transformative, “compound” AI:

Real-time agents, RAG and tool usage, model routing

How function calling works

1. Developer defines functions
2. User query
3. Model outputs structured text to call API
4. API call executed and results sent to model
5. Model summarizes results

```
{  
  "name": "get_stock_price",  
  "description": "Get a stock price",  
  "parameters": {  
    "type": "object",  
    "properties": {  
      "symbol": {  
        "type": "string",  
        "description": "The stock symbol",  
        "examples": ["GOOG"]  
      }  
    }  
  }  
}
```



Function calling is hard still

User: Plot and describe the stock movement of Nvidia and Apple over the last two weeks

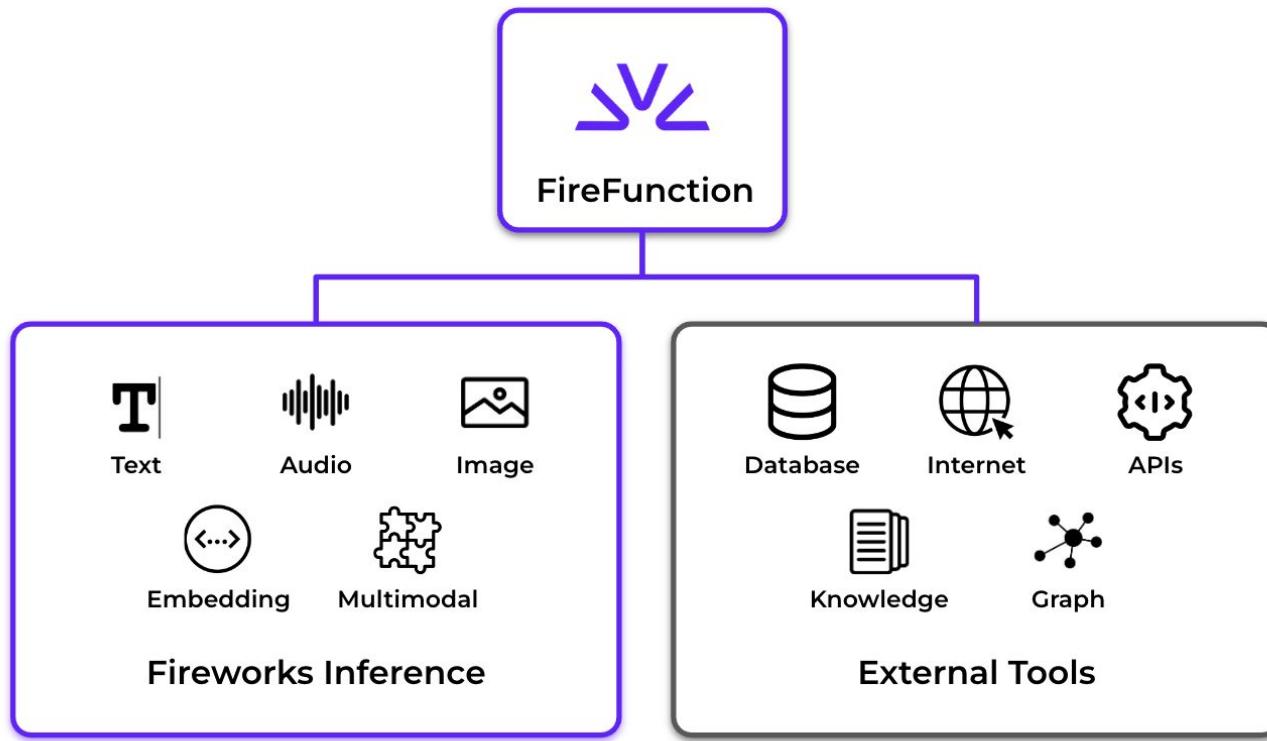
Assistant: [
 {
 "name": "get_stock_price",
 "arguments": {"ticker": "NVDA", "start_time": "2 weeks ago"}
 },
 {
 "name": "get_stock_price",
 "arguments": {"ticker": "AAPL", "start_time": "2 weeks ago"}
 }
]

Tool: {"NVDA": [120, 121, ...], "APPL": [192, 190, ...]}

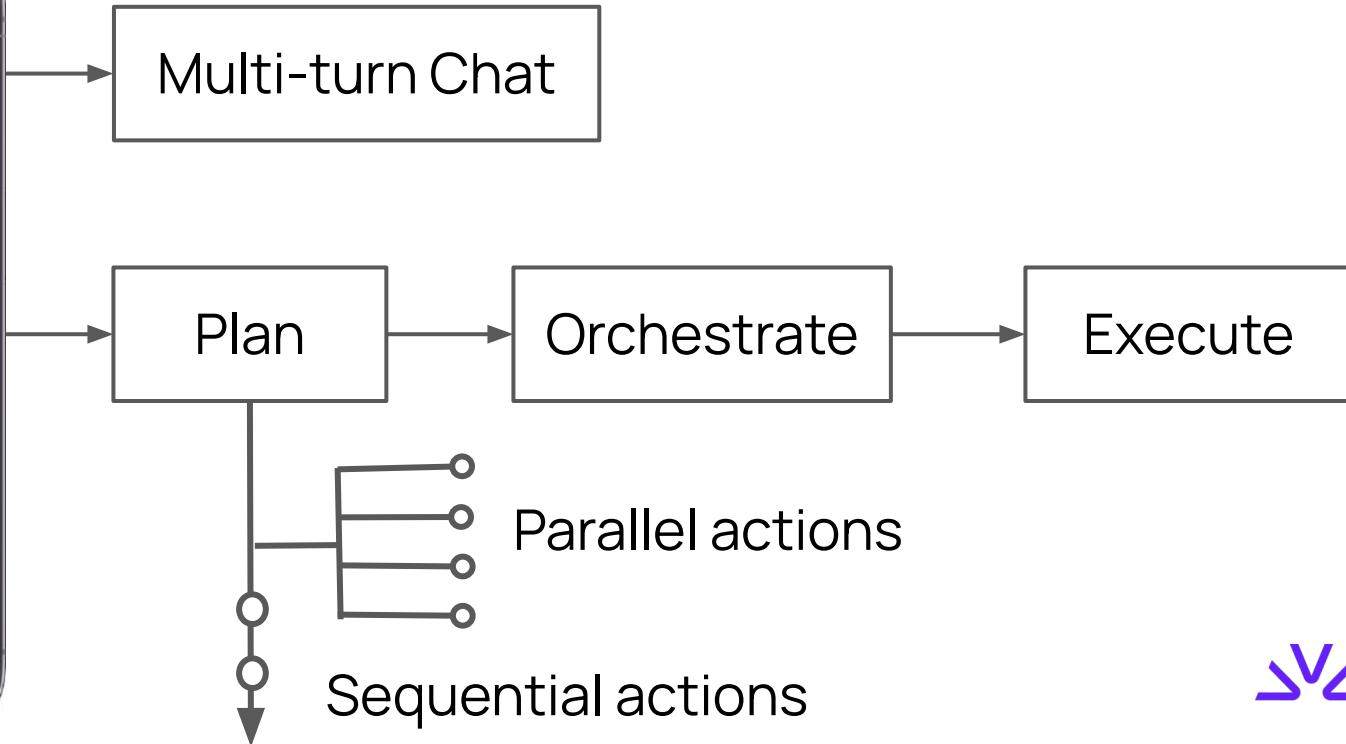
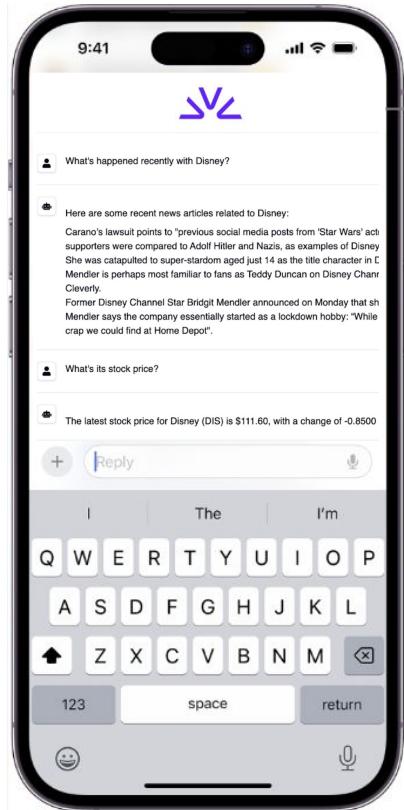
Assistant: {
 "name": "plot",
 "arguments": {"NVDA": [120, 121, ...], "APPL": [192, 190, ...]}
}



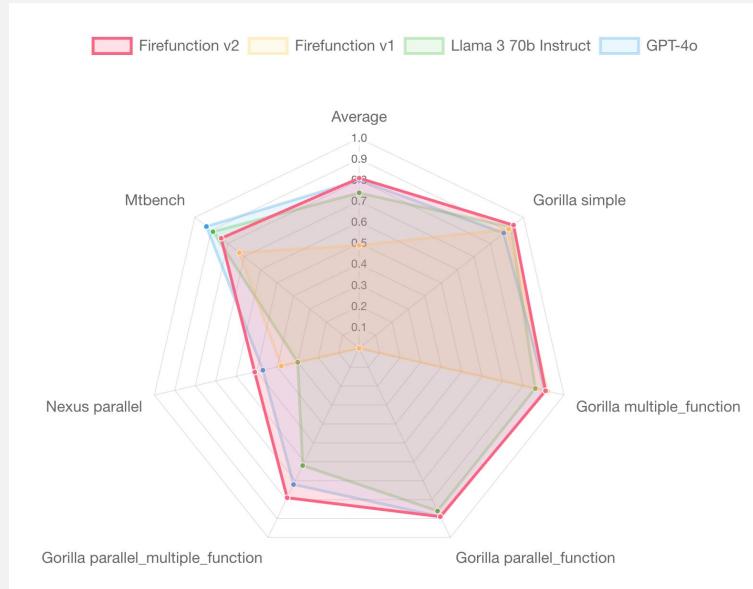
Fireworks Compound AI System



FireFunction - the Conductor and Executor



Firefunction-v2: GPT-4o performance at <10% cost and 2x speed



Leading model quality



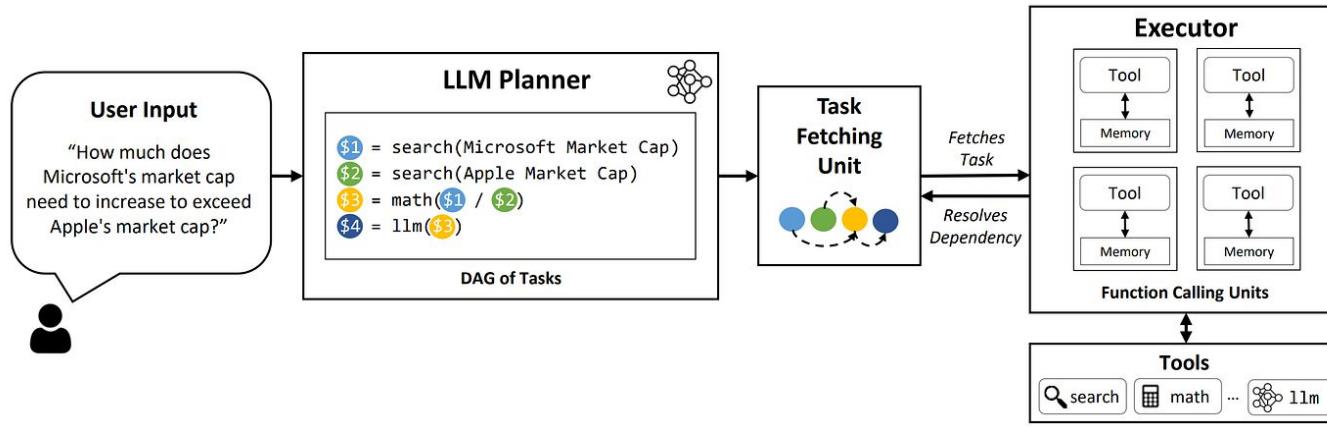
“Mind boggling” speed

Pattern 4: Fine-tuning

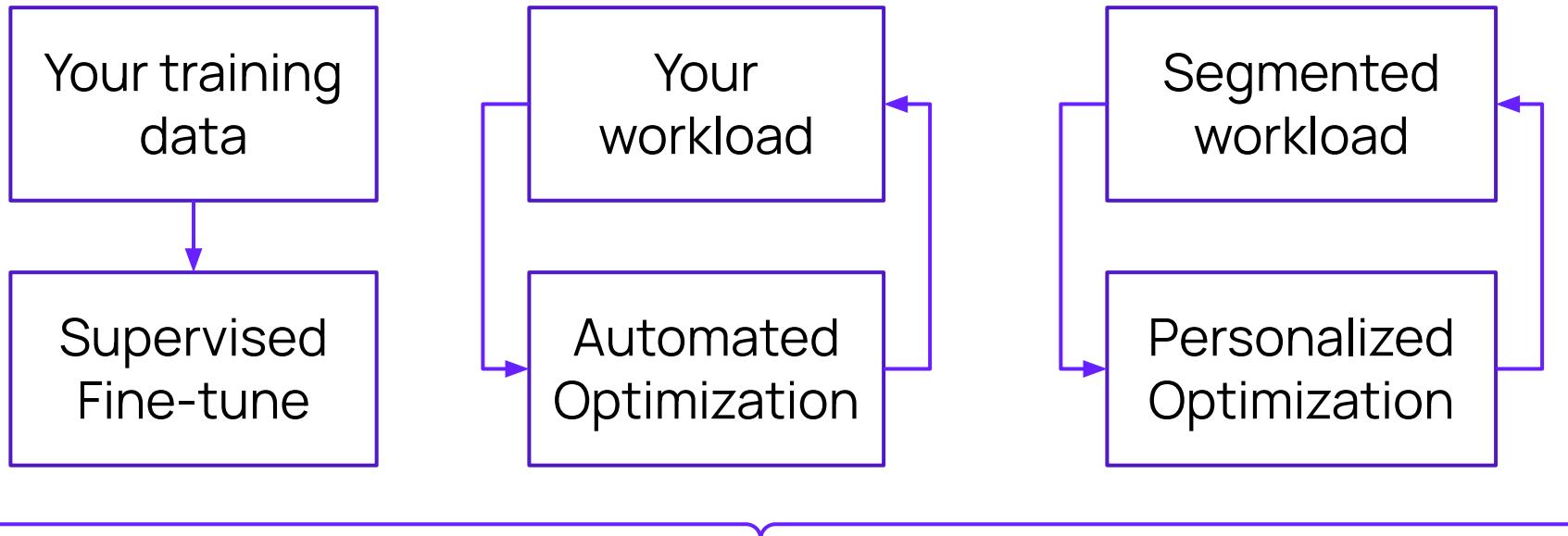


Section overview

- How to start fine-tuning a base LLM.
- Alternative approaches like prompt engineering.
- Discussion of pre-tuned models.



Fireworks Adaptation



Your own expert models for
differentiated value



Conclusion



The steps

- Check back on the Github repo: I'll be uploading new code; much of this will also be merged soon after in our cookbook repo.
 - Workshop URL: <https://bit.ly/SmallDataSF-FW>
 - Cookbook: <https://github.com/fw-ai/cookbook>
- If you create a project using Fireworks, let us know! Create a PR on the cookbook repo & share with us in Discord - we'd love to give you a shout-out, more credits (& potentially SWAG)!



Feel free to connect with me on:

-  LinkedIn: <https://www.linkedin.com/in/mikikobazeley/>
-  GitHub: <https://github.com/MMBazel>
-  Twitter: <https://twitter.com/BazeleyMikiko>
-  Substack: <https://mikikobazeley.substack.com/>
-  Youtube: https://www.youtube.com/@Miki_ML
- Discord!

