Open-Source Large Language Models in Radiology:
A Review and Tutorial for Practical Research and Clinical Deployment

Savage et al. Radiology 2025

A review

By Zineb El yamani
August 11, 2025

Open source VS Proprietary

Performance

Customizability

Cost

Licensing /innovation

Data security

Safety

Open source VS Proprietary

Performance P

Open source VS Proprietary

Customizability

Open-source:

you own the model

Proprietary:

Always under platform control

Risk of workflow disruption if a model version is deprecated

Open source VS Proprietary

Cost

Open-source:

Smaller models

No API costs (only hardware and maintenance expenses)

Proprietary:

Large parameter counts

Pay-per-token for every API request

Open source VS Proprietary

Open-source:

Encourages institutional and entrepreneurial innovation.

Proprietary:

- Some platforms allow revenue sharing, but always within the company's ecosystem.

Licensing /innovation

Open source VS Proprietary

Open-source

avoids sending sensitive data to third parties.

Security depends on local IT infrastructure, which may be weaker than big cloud providers.

Data security

Proprietary

Cloud vendors often have advanced, **HIPAA-compliant** security.

Open source VS Proprietary

Open-source

Less consistent testing for harmful outputs.

Developers often have fewer resources for adversarial robustness testing (prompt injection defenses).

Proprietary

Large-scale safety evaluation (red teaming, multiple safeguard layers).

Better resilience to adversarial prompts.

Safety

P

Open source VS Proprietary

Performance P

Customizability O

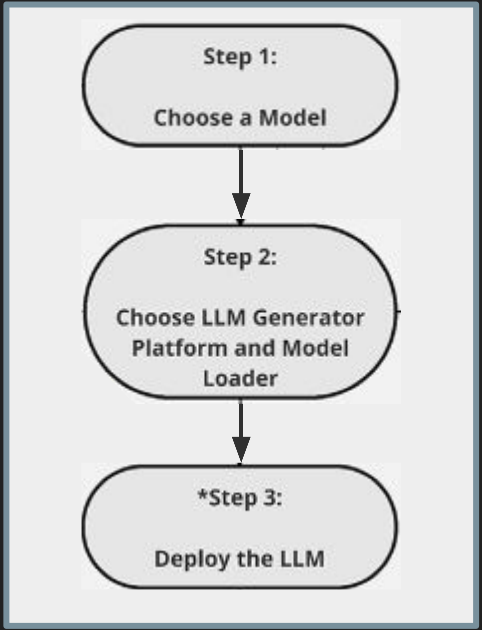
Cost O

Licensing /innovation O

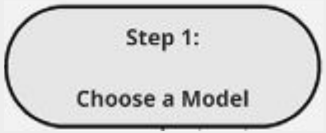
Data security

Safety P

I am a radiologist, how can I implement an open source LLM ?

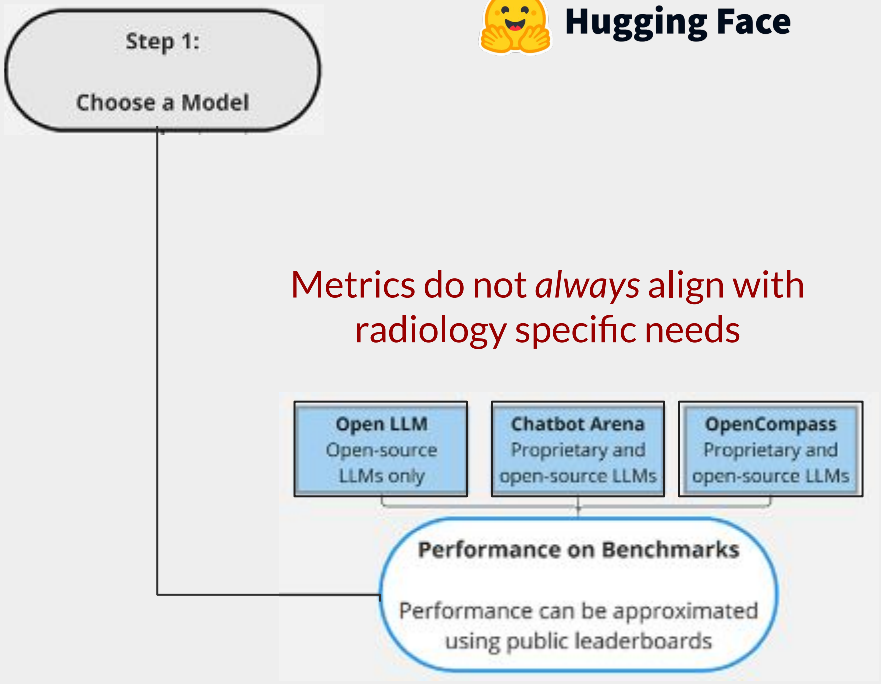


Hugging Face



Hugging Face

Metrics do not *always* align with radiology specific needs



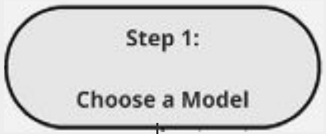
Further are needed !!

-complex multistep instructions
-Use case specific metrics
-Depend whether we consider end point to be fully automated or collaborative

RaLEs

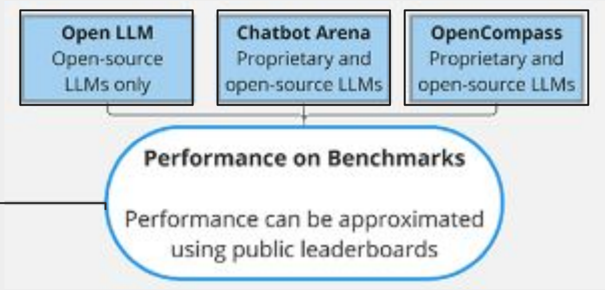
PubMedQA

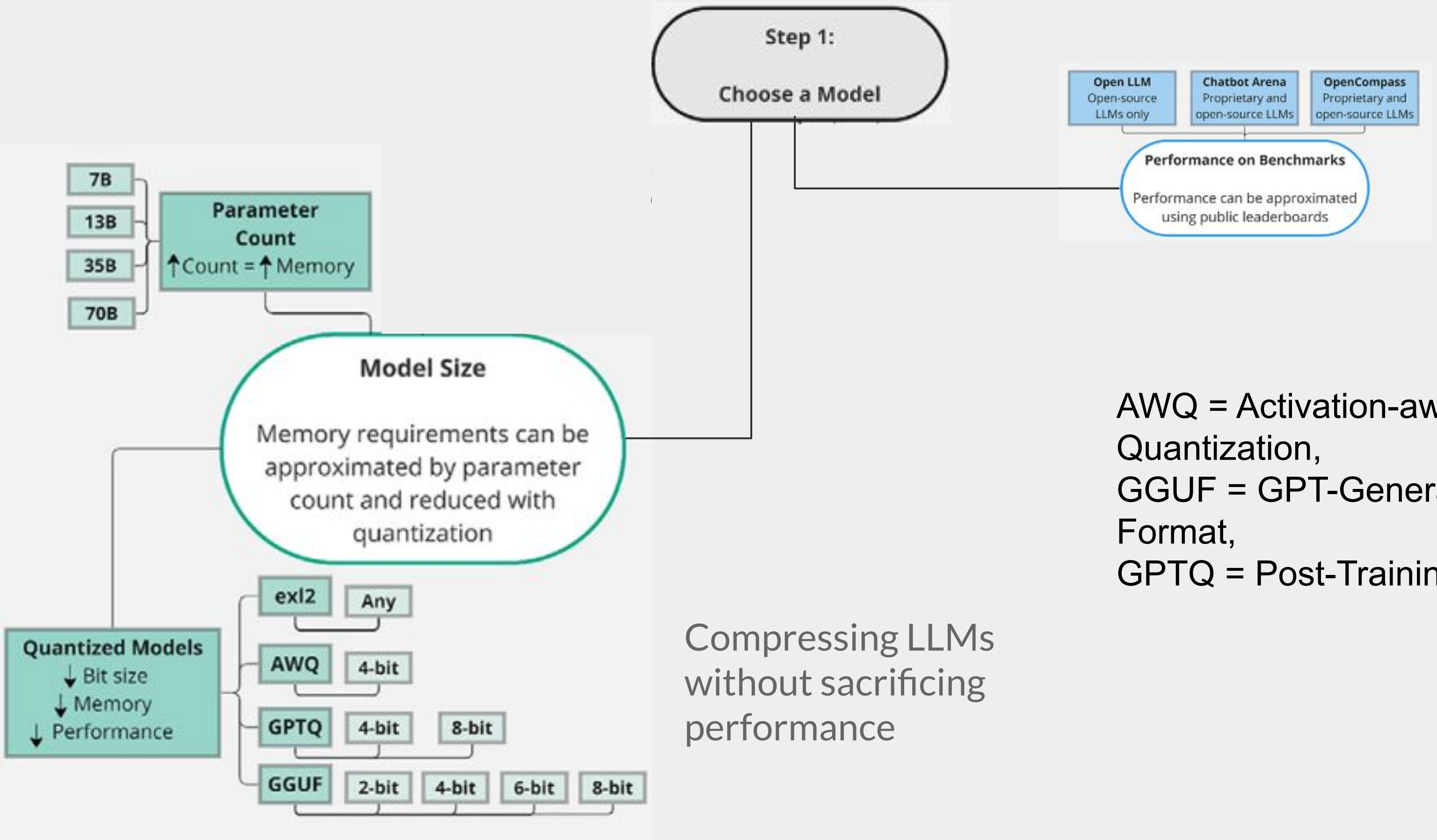
A Dataset for Biomedical Research Question Answering



Hugging Face

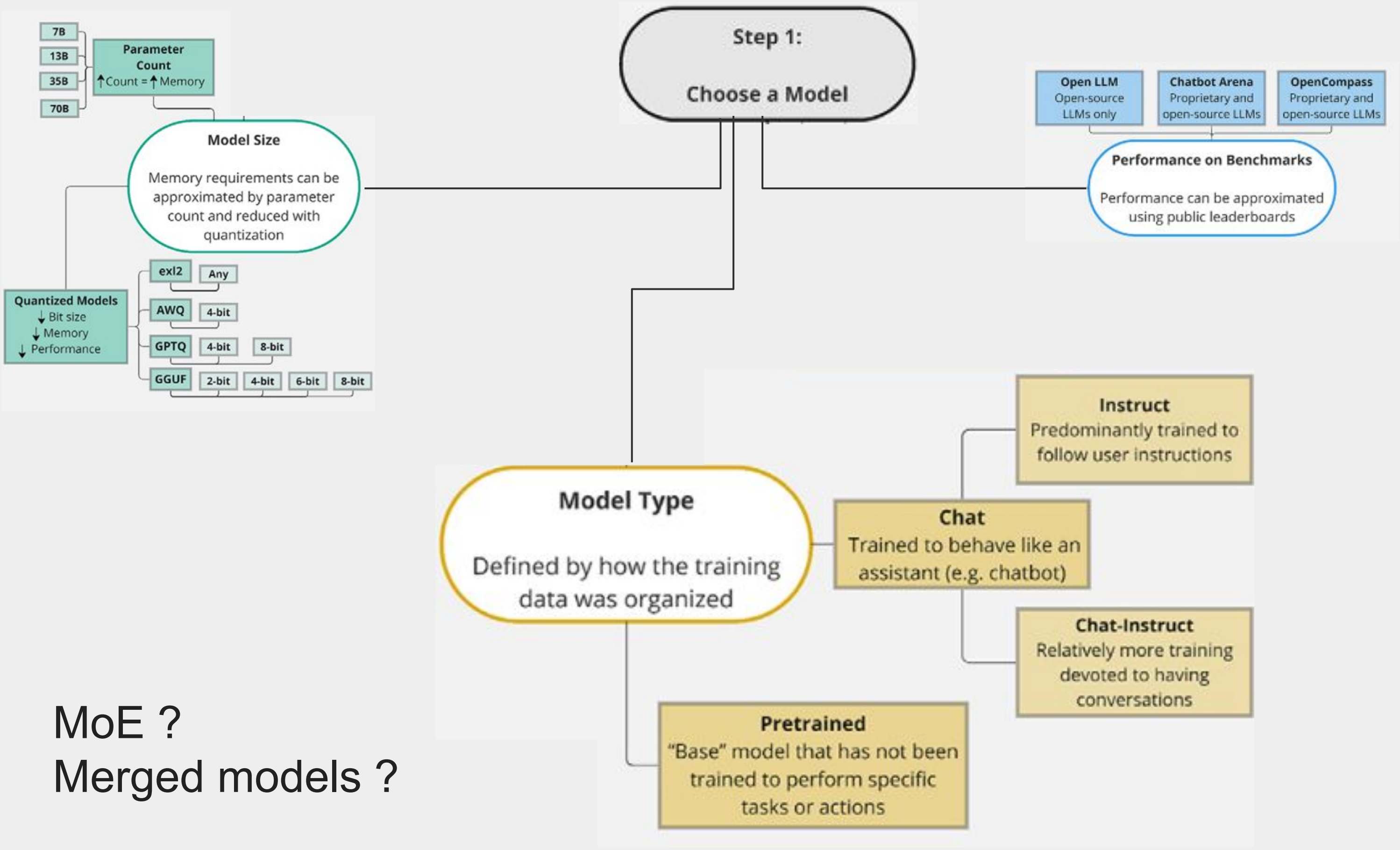
Metrics do not *always* align with radiology specific needs





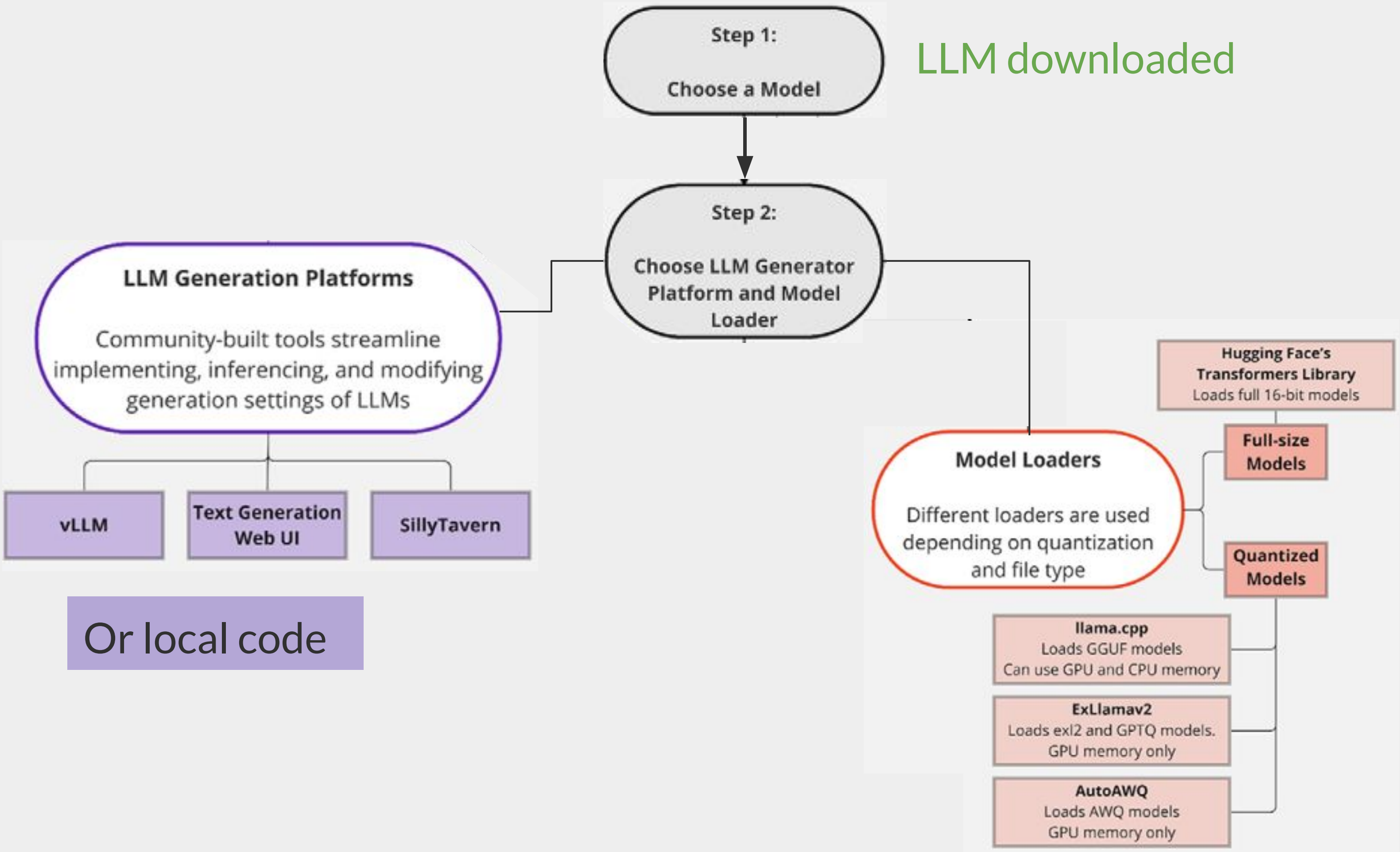
Compressing LLMs without sacrificing performance

AWQ = Activation-aware Weight Quantization,
GGUF = GPT-Generated Unified Format,
GPTQ = Post-Training Quantization



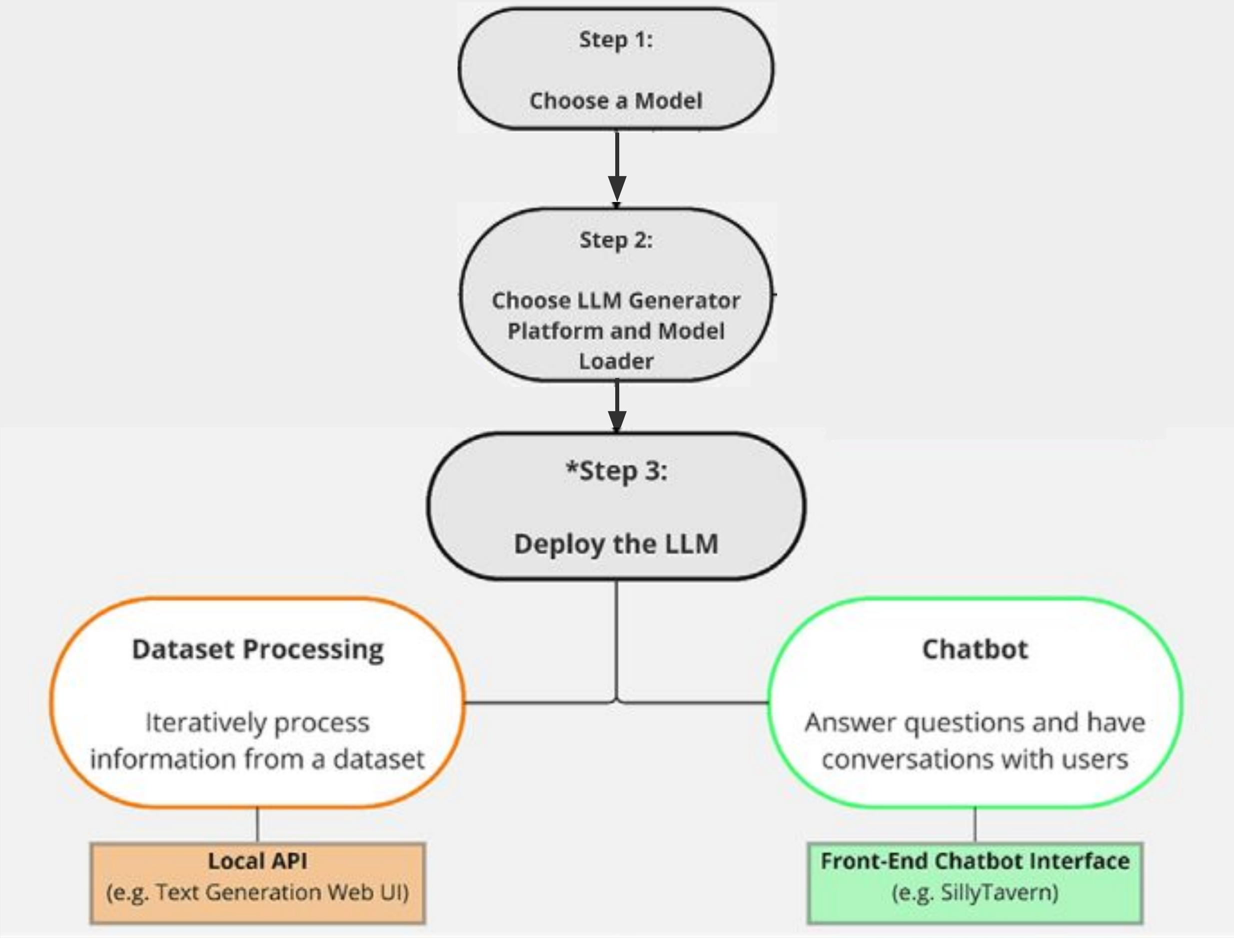
MoE ?

Merged models ?



LLM downloaded

Or local code


Step 1:
Choose a Model
Step 2:
Choose LLM Generator Platform and Model Loader
*Step 3:
Deploy the LLM
Dataset Processing
Iteratively process information from a dataset
Local API
(e.g. Text Generation Web UI)
Chatbot
Answer questions and have conversations with users
Front-End Chatbot Interface
(e.g. SillyTavern)

Troubleshooting Performance issues

Prompt engineering

Retrieval-augmented Generation

Fine-tuning

Prompt engineering

token-based solutions

Problem :

LLMs may exhibit deficiencies in complex reasoning, defined as low performance on tasks that require **multistep reasoning** (eg, generating a differential diagnosis)

Common techniques:

- ➔ Chain-of-Thought (CoT): Tell the model to "think step by step," so it breaks reasoning into intermediate steps.
- ➔ Reflexion: The model simulates an *evaluator* that critiques its own first answer, then revises it based on that feedback.
- ➔ Few-shot prompting: Give a few solved examples in the prompt before the real question.

Retrieval-augmented Generation

Problem :

LLMs can have an **insufficient** knowledge base that can potentially lead to hallucinations. (eg, constantly changing medical guidelines)

Solution:

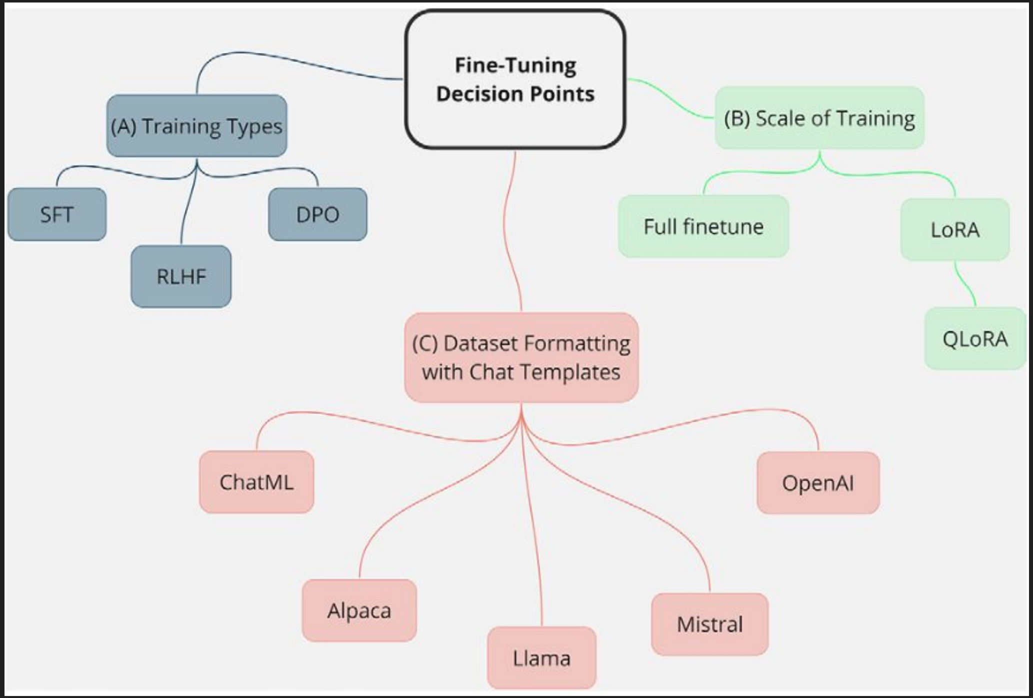
➔ Supplement the input prompt with information from other data source without the need for fine-tuning.

Fine-tuning

Problem :
LLMs can exhibit **poor performance** in instruction following.

Solution:

➔ Retrain the model with additional domain-specific data so it internalizes new knowledge or skills.



Fine-tuning

Training methods: *depend on the complexity and breadth of the desired task*

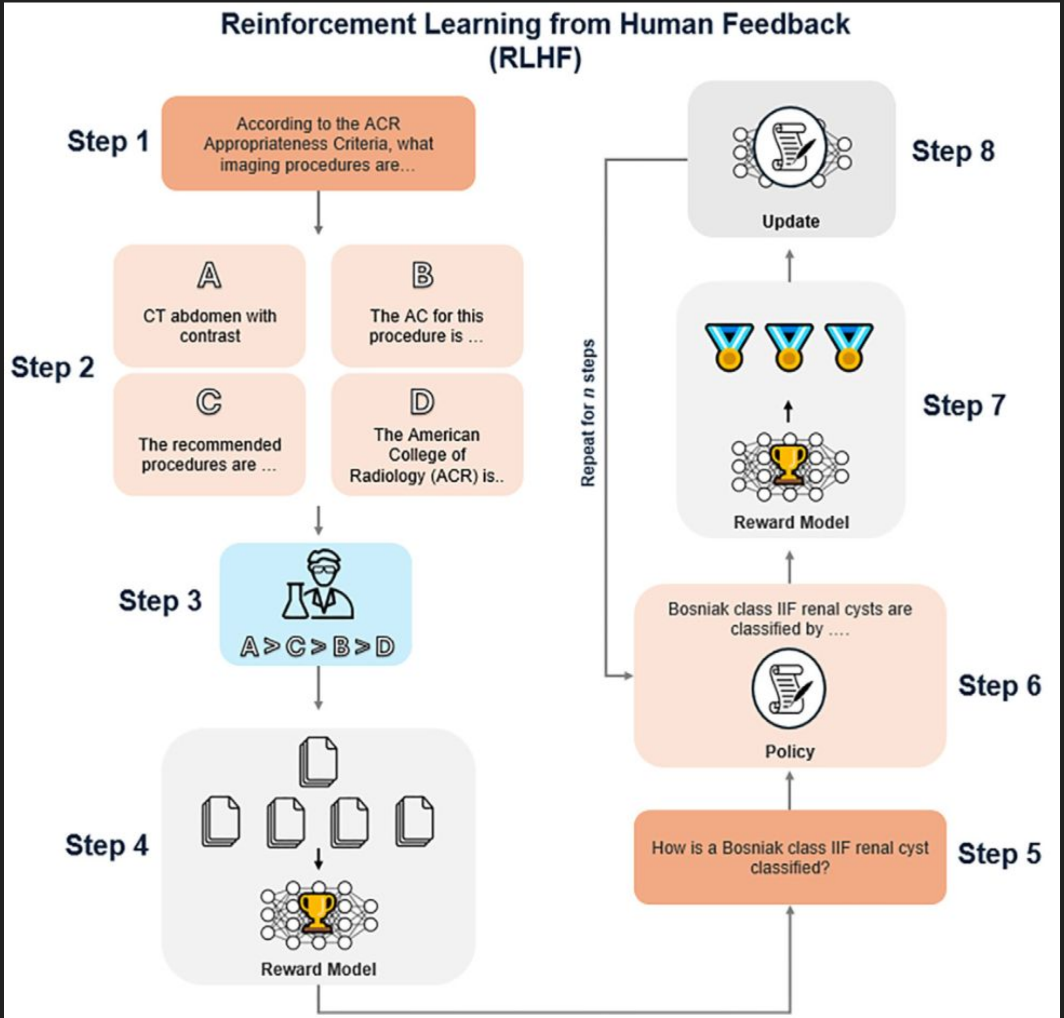
SFT (Supervised Fine-Tuning): Train on prompt–response pairs.

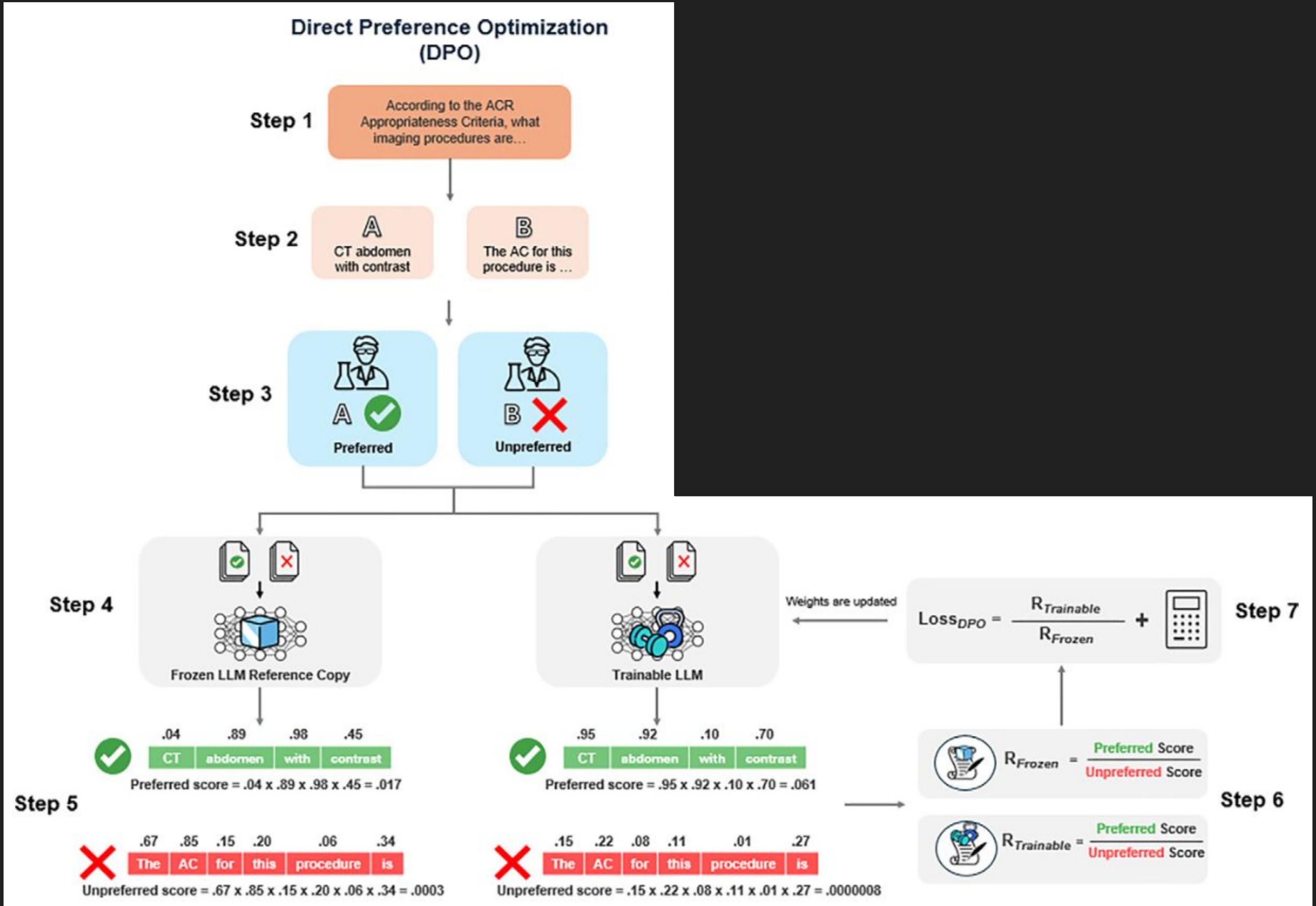
Good for well-defined tasks with a narrow range of correct answers

RLHF (Reinforcement Learning from Human Feedback): Humans rank outputs, a reward model learns preferences, and the LLM adapts to maximize that reward.

DPO (Direct Preference Optimization): Like RLHF but skips the reward model

Simpler and needs fewer examples.


Reinforcement Learning from Human Feedback (RLHF)
Step 1
According to the ACR Appropriateness Criteria, what imaging procedures are…
Step 2
A
CT abdomen with contrast
B
The AC for this procedure is …
C
The recommended procedures are …
D
The American College of Radiology (ACR) is..
Step 3
A > C > B > D
Step 4
Reward Model
Step 5
How is a Bosniak class IIF renal cyst classified?
Step 6
Bosniak class IIF renal cysts are classified by ….
Policy
Step 7
Reward Model
Step 8
Update
Repeat for n steps


Direct Preference Optimization
(DPO)
Step 1
According to the ACR Appropriateness Criteria, what imaging procedures are…
Step 2
A
CT abdomen with contrast
B
The AC for this procedure is …
Step 3
A
Preferred
B
Unpreferred
Step 4
Frozen LLM Reference Copy
Trainable LLM
Weights are updated
Loss_DPO = R_Trainable / R_Frozen +
Step 7
Step 5
.04 .89 .98 .45
CT abdomen with contrast
Preferred score = .04 x .89 x .98 x .45 = .017
.67 .85 .15 .20 .06 .34
The AC for this procedure is
Unpreferred score = .67 x .85 x .15 x .20 x .06 x .34 = .0003
.95 .92 .10 .70
CT abdomen with contrast
Preferred score = .95 x .92 x .10 x .70 = .061
.15 .22 .08 .11 .01 .27
The AC for this procedure is
Unpreferred score = .15 x .22 x .08 x .11 x .01 x .27 = .0000008
R_Frozen = Preferred Score / Unpreferred Score
R_Trainable = Preferred Score / Unpreferred Score
Step 6

Fine-tuning

Training scale: *depend on the reasoning ability required and the computational resources of the user*

Full fine-tune: Adjust all model parameters *(best performance but expensive)*.

LoRA (Low-Rank Adaptation): Only adjust small parameter subsets *(much cheaper in memory and time)*.

QLoRA: LoRA + quantized LLM *(even lower resource usage)*.

I am a radiologist, how can I implement an open source LLM ?

Define the Problem First

➔ What is the *exact* use case? Clinical task? Research? Administrative?

➔ What's the measurable outcome? Accuracy, speed, cost savings, reduced workload, patient safety?

Regulatory & Risk Context

➔ Will this be used in clinical care or just for research?

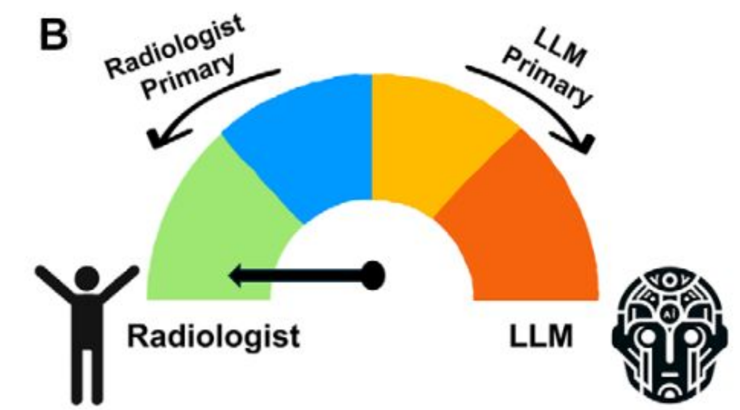
➔ Is a human-in-the-loop required?

➔ Does the output have direct patient impact?

➔ What are the privacy laws that apply (HIPAA, …)?

➔ What's the risk if the model hallucinates or makes an error?

➔ Implement post-deployment surveillance for safety.



Cost & Resource Planning

Do you need the best performance immediately, with minimal setup?

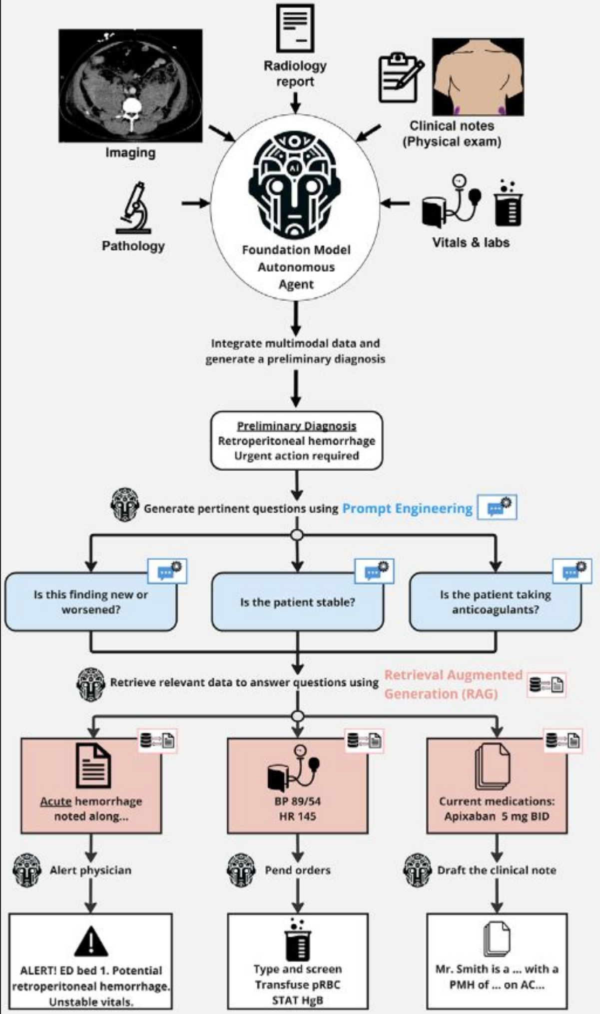
➔ **Proprietary models:** token-based API costs (input + output).

Do you need full control over the model?

➔ **Open-source models:** hardware cost (GPUs, cloud compute), IT staff time, energy consumption.

Do you plan for scaling ?

➔ more users = more compute or higher API spend.


Radiology report
Imaging
Clinical notes (Physical exam)
Pathology
Foundation Model Autonomous Agent
Vitals & labs
Integrate multimodal data and generate a preliminary diagnosis
Preliminary Diagnosis
Retroperitoneal hemorrhage
Urgent action required
Generate pertinent questions using Prompt Engineering
Is this finding new or worsened?
Is the patient stable?
Is the patient taking anticoagulants?
Retrieve relevant data to answer questions using Retrieval Augmented Generation (RAG)
Acute hemorrhage noted along...
BP 89/54
HR 145
Current medications: Apixaban 5 mg BID
Alert physician
Pend orders
Draft the clinical note
ALERT! ED bed 1. Potential retroperitoneal hemorrhage. Unstable vitals.
Type and screen
Transfuse pRBC
STAT HgB
Mr. Smith is a ... with a PMH of ... on AC...