

✦ COMPARISON · RAG SYSTEMS

RAG vs Fine-Tuning

Two ways to make a model work on your domain — one changes what it reads, the other changes how it behaves. When to use each, and when to use both.

FORMAT	LEVEL	READ TIME	UPDATED
Comparison	Intermediate	10 min	Oct 2026

OVERVIEW

What this comparison covers

Retrieval-augmented generation gives a model the right information at question time. Fine-tuning changes the model itself through additional training. They are often framed as competitors, but they solve different problems: RAG is the tool for knowledge that changes and must be cited; fine-tuning is the tool for consistent behavior, format and efficiency. This comparison sets them side by side, gives a decision guide, and describes the combined approach many mature systems use.

KEY TAKEAWAYS

- RAG supplies knowledge; fine-tuning shapes behavior. Match the tool to the gap.
- If facts change or answers must cite sources, use retrieval.
- If the model knows enough but behaves inconsistently, or is too costly at volume, consider fine-tuning.
- Try strong prompting first — it often closes the gap without either.
- Many production systems combine both: a tuned model that is good at using retrieved context.

WHO IT'S FOR

- CTOs and engineering leaders
- AI engineers choosing an architecture
- Product owners scoping domain assistants
- Data and compliance teams

HOW TO USE IT

Start with the table in section 01, then take your use case through the decision cards in section 02 and the checklist in section 05.

CONTENTS

01 How they compare
02 When to choose which
03 Using both together
04 Cost and effort profile
05 Decision checklist

01 / SIDE BY SIDE

How they compare

RAG changes the model's input; fine-tuning changes the model's weights. Almost every practical difference follows from that.

DIMENSION	RAG	FINE-TUNING
What it changes	The context the model reads	The model's weights
Best at	Facts, documents, current data	Format, style, task behavior
Freshness	Update the index; live immediately	Requires retraining
Citations	Natural — answer points to sources	Not available from the weights
Access control	Filter per user at retrieval	Whatever is trained is available to every user
Upfront effort	Ingestion, chunking, search pipeline	Labeled data, training runs, evals
Per-request cost	Longer prompts from retrieved context	Shorter prompts; tuned-model pricing or hosting
Failure mode	Wrong or missing chunks retrieved	Confidently wrong or outdated outputs
Maintenance	Keep connectors and index in sync	Retrain on drift and base-model upgrades

RAG

- Knowledge stays outside the model
- Updates in minutes, not training cycles
- Answers can cite and be checked
- Quality depends on retrieval

FINE-TUNING

- Behavior baked into the model
- Consistent format with short prompts
- Can let a smaller model do the job
- Poor at facts that change

✦ 02 / DECISION

When to choose which

Identify the gap first: is the model missing information, or is it mishandling a task it has the information for?

CHOOSE RAG

Answers depend on your documents

Policies, manuals, contracts, tickets — content the base model has never seen.

CHOOSE RAG

Content changes often

Prices, procedures and product details that would be stale after training.

CHOOSE RAG

Users have different access

Retrieval can filter per user; trained weights cannot.

CHOOSE FINE-TUNING

Behavior is inconsistent

Format, tone or classification drift despite a well-engineered prompt.

CHOOSE FINE-TUNING

Volume makes cost matter

A tuned smaller model could replace a large model with a long prompt.

CHOOSE FINE-TUNING

Specialized language

Domain notation or jargon the base model handles poorly.

Decision sequence

01 **Write the best prompt you can**

With clear instructions and examples. Measure it on an eval set.

02 **If it lacks knowledge, add retrieval**

Build RAG and measure retrieval and answer quality separately.

03 **If behavior is still off, consider tuning**

Fine-tune for the remaining behavior gap, keeping retrieval for facts.

04 **Compare against the baseline**

Ship the more complex option only if evals show it is better at acceptable cost.

03 / HYBRID

Using both together

The approaches compose well. A common pattern is retrieval for knowledge plus a model tuned to use retrieved context well and answer in the required format.

01 User question → 02 Hybrid retrieval + rerank → 03 Permission-filtered context → 04 Tuned model answers → 05 Citations + validation

HYBRID

Tune for grounding

Train on examples that answer only from supplied context and decline when it is insufficient.

HYBRID

Tune the retriever

Fine-tune an embedding model or reranker on your domain to improve what RAG finds.

HYBRID

Tune small helpers

Use small tuned models for query rewriting or routing inside a RAG pipeline.

FIELD NOTE

Do not train facts you also retrieve

If a fact is in both the weights and the index, outdated trained knowledge can override fresh retrieved context. Keep facts in retrieval and train behavior.

04 / COSTS

Cost and effort profile

Both carry ongoing cost. Plan for the whole lifecycle, not just the build.

PHASE	RAG	FINE-TUNING
Build	Connectors, parsing, index, retrieval tuning	Data collection, labeling, training runs
Run	Search infrastructure and longer prompts	Tuned-model pricing or GPU hosting
Change	Re-index content; adjust retrieval	Rebuild dataset and retrain

PHASE	RAG	FINE-TUNING
Evaluate	Retrieval recall plus answer quality	Task metrics plus regression checks
Own	Data pipeline and index freshness	Training pipeline and model versions

✦ 05 / CHECKLIST

Decision checklist

Work through this for your use case before committing to either approach.

THE GAP

- [ ] A strong prompted baseline has been measured
- [ ] Failures are classified as missing knowledge or wrong behavior
- [ ] An eval set exists that reflects real questions

KNOWLEDGE

- [ ] How often the source content changes is known
- [ ] Citation and audit needs are stated
- [ ] Per-user access rules are understood

BEHAVIOR AND COST

- [ ] Required output format and tone are specified
- [ ] Request volume and cost targets are estimated
- [ ] Labeled examples are available or can be produced

OWNERSHIP

- [ ] An owner for index freshness or retraining is named
- [ ] Base-model upgrade plan is agreed
- [ ] Rollback path to the simpler approach exists

> Retrieve what the model needs to know; train how it should behave.
>
> SPLENDENSLABS ENGINEERING PRINCIPLE

✦ GET IN TOUCH

Let's Talk

We build production RAG systems and fine-tuned models, and help teams choose — with evals, not opinions — which their use case needs.

- ✦ Baseline evaluation of your use case
- ✦ RAG, fine-tuning or hybrid recommendation
- ✦ Architecture and cost model
- ✦ A measured pilot against your real questions

BOOK AN AI STRATEGY SESSION →

EMAIL
hello@splendenslabs.com

WEB
splendenslabs.com

OFFICE
Sona Incubation Foundation, Salem