RAG 101 : Fine-tuning ou RAG?

RAG 101: Fine-tuning ou RAG? Na série NLP 101, acompanhamos como as máquinas passaram da contagem de palavras aos embeddings, RNNs, Attention, Transformers e LLMs. Agora começa…

2026-07-15



Figure 1: Capa do artigo

RAG 101: Fine-tuning ou RAG? Na série NLP 101, acompanhamos como as máquinas passaram da contagem de palavras aos embeddings, RNNs, Attention, Transformers e LLMs. Agora começa uma nova etapa. Na RAG 101, vamos olhar para a engenharia construída ao redor desses modelos: como buscar conhecimento, preservar contexto e gerar respostas apoiadas em fontes. Vem comigo!

Antes de falar de Naive RAG, bancos vetoriais, agentes e grafos, precisamos resolver uma dúvida que aparece em quase todo projeto de IA: quando vale a pena fazer fine-tuning? E quando o modelo deveria consultar uma fonte externa? a resposta mais simples é que as duas abordagens atuam em partes diferentes do sistema e elas tem propositos diferentes e podem ser complementares.

Thread Pense em uma pessoa começando a trabalhar em um restaurante. O fine-tuning ensina como ela deve atender: o tom de voz, o vocabulário, os procedimentos e o formato dos pedidos. O RAG entrega o cardápio atualizado para consulta. Um molda a forma de trabalhar, o outro fornece informação durante a tarefa.

No fine-tuning, exemplos são usados para ajustar os pesos do modelo. Ele pode aprender a: responder em um formato específico, seguir uma terminologia técnica, classificar textos, gerar SQL em um padrão interno, executar melhor uma tarefa de domínio Esse aprendizado fica incorporado aos parâmetros. Só que os pesos funcionam como uma memória estática. Imagine treinar um modelo com o catálogo de uma empresa. na semana seguinte, os preços mudam, um item sai de estoque e outro é lançado.

Para incorporar essas alterações aos pesos, seria necessário preparar dados e executar outro ciclo de treinamento.

© Artemisia Weyl 2024–2026 — Todos os direitos reservados.



Figure 2: Imagem original inserida na publicação 1/8

No RAG, os pesos permanecem iguais. Quando recebe uma pergunta, o sistema busca informações em documentos, bancos, APIs ou outros repositórios. As evidências relevantes são colocadas na janela de contexto, e a LLM produz a resposta a partir do material recuperado. É como fazer uma prova com consulta.

Isso torna o RAG especialmente útil para informações que mudam com frequência: preços,normas, documentação interna, relatórios, políticas da empresa, dados operacionais A fonte pode ser atualizada sem treinar o modelo novamente, e também fica mais fácil mostrar de onde cada informação veio. Essa flexibilidade traz custos Cada pergunta pode exigir busca e mais tokens no prompt, aumentando a latência e o custo por consulta. No fine-tuning, o investimento maior ocorre antes, durante o retreino

Uma armadilha comum é usar fine-tuning para tentar armazenar fatos que mudam. O modelo pode memorizar parte do conteúdo, mas isso oferece pouca garantia de atualização, rastreabilidade ou recuperação precisa. Fine-tuning costuma funcionar melhor para padrões de comportamento. RAG é especialmente útil quando a resposta depende de evidências externas. Na prática, muitos sistemas combinam as duas abordagens.

O fine-tuning ajuda o modelo a entender o domínio, seguir formatos e executar tarefas específicas e o RAG fornece os documentos e dados necessários naquele momento. No próximo post vamos falar de Naive RAG, a famosa arquitetura de PDF picado, top-k e fé. Gostou, comenta

---

© Artemisia Weyl 2024–2026 — Todos os direitos reservados.