🤔 FAQ + Is Fine-tuning Right For Me?

If you're stuck on if fine-tuning is right for you, see here!

Understand Fine-tuning

Fine-tuning an LLM customizes its behavior, enhances domain knowledge, and optimizes performance for specific tasks. By fine-tuning a pre-trained model (e.g. Llama-3.1-8B) on a specialized dataset, you can:

- **Update Knowledge**: Introduce new domain-specific information.
- **Customize Behavior**: Adjust the model’s tone, personality, or response style.
- **Optimize for Tasks**: Improve accuracy and relevance for specific use cases.

You can think of a fine-tuned model as a specialized agent designed to do specific tasks more effectively and efficiently. When deciding between using Retrieval-Augmented Generation (RAG) or fine-tuning, it's important to note that fine-tuning can replicate RAG's functionalities and not vice versa. Instead of using one or the other, we recommend people to use both which greatly increases accuracy, useability and reduces hallucinations.

Example usecases:

- Train LLM to predict if a headline impacts a company positively or negatively.
- Use historical customer interactions for more accurate and custom responses.
- Fine-tune LLM on legal texts for contract analysis, case law research, and compliance

Benefits of Fine-tuning

1. **Fine-tuning can do everything RAG can, but RAG can't**

Fine-tuning can replicate RAG's functionality by embedding external knowledge directly into the model during training, allowing it to perform tasks like answering niche questions or summarizing documents without relying on external systems. Fine-tuning can also integrate context and patterns into the model, mimicking retrieval behavior.

2. **Task-Specific Mastery**
   Fine-tuning embeds deep knowledge of a domain or task directly into the model, enabling it to handle structured, repetitive, or nuanced queries with high accuracy - something RAG alone cannot achieve.
3. **Independence from Retrieval**
   A fine-tuned model operates effectively without external data, ensuring seamless performance even when the retrieval system fails or the knowledge base is incomplete.
4. **Faster Inference**
   Fine-tuned models provide direct responses without needing a retrieval step, making them ideal for scenarios where speed is critical.
5. **Custom Behavior and Tone**
   Fine-tuning allows for precise control over how the model behaves and communicates, ensuring alignment with brand voice, regulatory requirements, or other constraints.
6. **Fallback Robustness**
   In combined systems, the fine-tuned model ensures a baseline level of reliable task performance, even if the RAG system retrieves irrelevant or incomplete information.

Does Fine-tuning Add New Knowledge to my Model?

Yes, absolutely! A common misconception is that fine-tuning does not introduce new knowledge into a model, but that's simply not true. In fact, the very purpose of fine-tuning is to teach the model entirely new concepts or knowledge as long as your dataset contains the relevant information. Fine-tuning enables the model to learn from scratch when presented with the specific data.

Does RAG perform better than Fine-tuning?

Another widespread misconception is that RAG consistently outperforms fine-tuning on benchmarks. This is incorrect. When fine-tuning is done properly, it often achieves superior results compared to RAG. Statements to the contrary often stem from improper implementation - such as misconfiguring LoRA parameters or a general lack of experience with fine-tuning.

Unsloth takes care of these complexities by automatically selecting the best parameter configurations for you. All you need is a good-quality dataset, and you'll get a fine-tuned model that performs to the best of its ability.

Combining RAG + Fine-tuning

That's why we suggest users to combine both RAG and fine-tuning together rather than using one or the other. RAG enhances adaptability by dynamically accessing external knowledge, while fine-tuning fortifies the system's core expertise, ensuring it performs reliably without over-relying on retrieval. Moreover, fine-tuning empowers the model to interpret and integrate retrieved information more effectively, producing seamless and contextually accurate responses.

Why should you combine RAG & fine-tuning?

- **Task-Specific Expertise**: Fine-tuning excels at specific tasks, while RAG dynamically retrieves up-to-date or external knowledge. Together, they handle both core and context-specific needs.
- **Adaptability**: Fine-tuned models provide robustness when retrieval fails, and RAG enables the system to stay current without constant re-training.
- **Efficiency**: Fine-tuning establishes a baseline, while RAG reduces the need for exhaustive training by handling dynamic details.

LoRA vs QLoRA

- **LoRA:** Fine-tunes small, trainable matrices in 16-bit without updating all model weights.
- **QLoRA:** Combines LoRA with 4-bit quantization to handle very large models with minimal resources.

We recommend starting with QLoRA, as it is one of the most accessible and effective methods for training models. Thanks to Unsloth's dynamic 4-bit quants, the accuracy loss for QLoRA compared to LoRA is now largely recovered.

Remember, experimentation is key

There’s no single "best" way to approach fine-tuning - only best practices that serve as general guidelines. However, these may not always be ideal for your specific dataset or use case. That’s why we encourage continuous experimentation to find what works best for your unique needs.

A great place to start is with QLoRA (4-bit), which offers a highly efficient, resource-friendly way to explore fine-tuning without heavy computational costs.

Is Fine-tuning Expensive?

Not at all! While full fine-tuning or pretraining can be costly, these are rarely necessary. In most cases, LoRA or QLoRA fine-tuning can be done for minimal cost. In fact, with Unsloth’s free notebooks for Colab or Kaggle, you can fine-tune models without spending a dime. Better yet, you can even fine-tune locally on your own device.

Previous
Unsloth Requirements

Next
What Model Should I Use?

Last updated 9 days ago

Was this helpful?