Local LLM fine-tuning: a practical example

Daniele Giunta

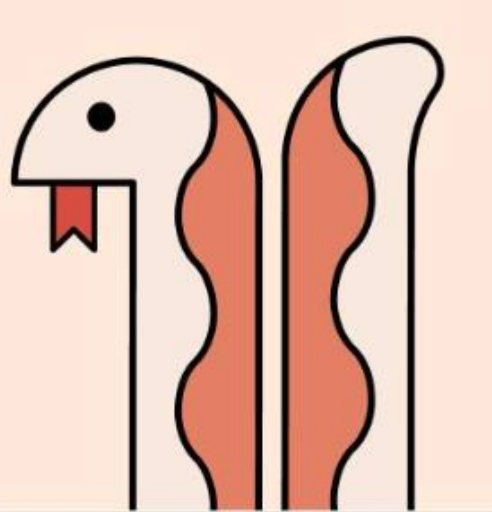
AI Engineer @ ELIS Innovation Hub

https://www.linkedin.com/in/daniele-g-dr16/

Bologna 29/05/2025

Agenda

- Why local LLM?
- What is Fine-Tuning?
- From theory to practice!
- Key Takeaways



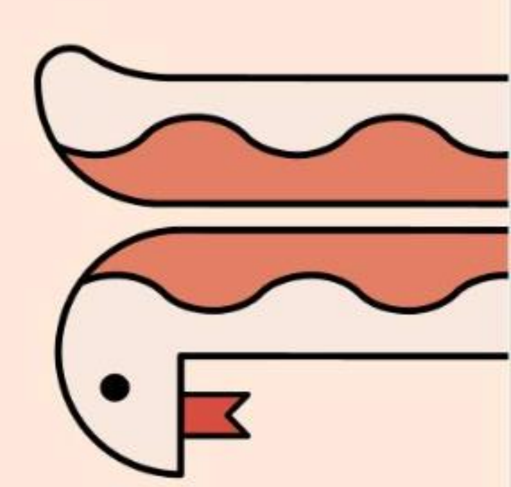
Why local LLM?

Overcome main **limits** of cloud-based LLMs!

1. **Privacy**: your data stays in your hands

1. **Customization**: adapt to your company/domain

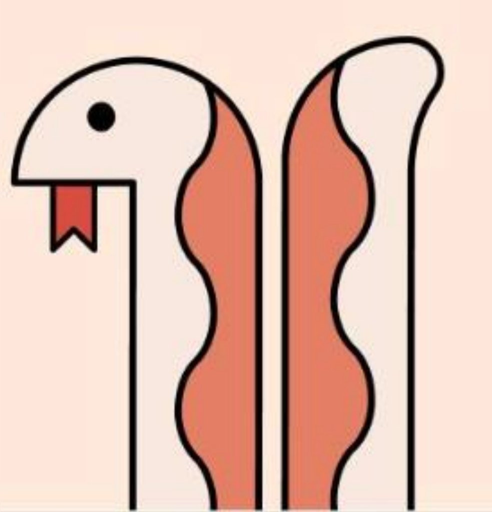
1. **Cost & speed**: avoid cloud/usage fees, faster response times



What is “Fine-Tuning”?

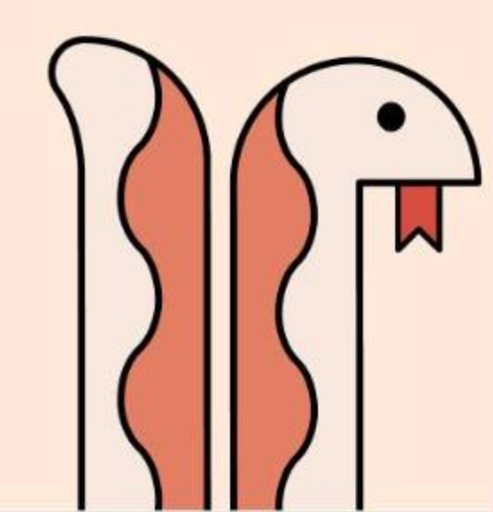
- Take a pre-trained LLM and **train** it **further**
- Use **supervised learning** with input-output pairs
- Example: creating a **customer support chatbot** for a retail company.

→ We make the model **expert** in **our context**!



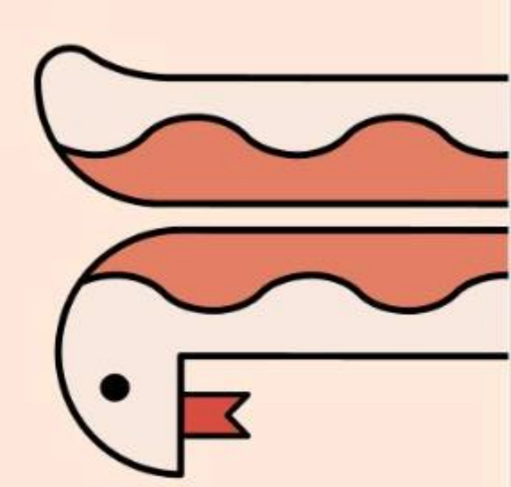
From theory to practice!

- Jupyter Notebook
- Google Colab
- Meta’s Llama 3.2
- Unsloth + HuggingFace
- AnythingLLM



Key Takeaways

- Local LLM fine-tuning is **practical** and **accessible**
- Tools like **Unsloth** + **HuggingFace** make it **easy**
- Great for **privacy**, **customization**, and **control**
- Fine-tuning isn’t as hard as it sounds—But **good data** is **crucial**, and it’s often the **hardest part**!



Thank you!

Daniele Giunta

AI Engineer @ ELIS Innovation Hub

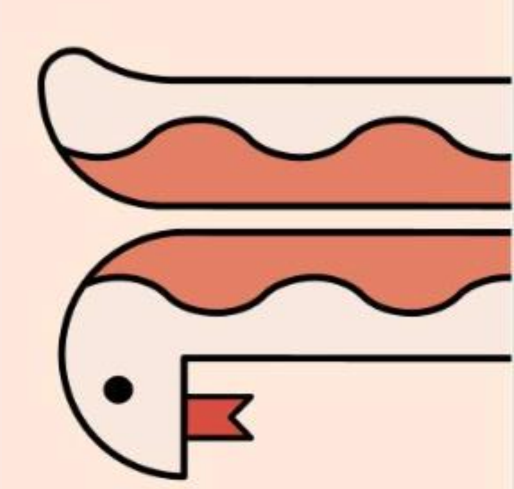
https://www.linkedin.com/in/daniele-g-dr16/

Bologna 29/05/2025

References

Useful links:

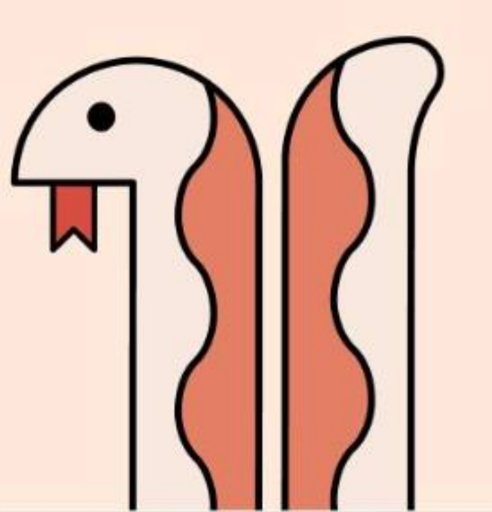
- http://github.com/unslothai/unsloth
- https://huggingface.co/blog/unsloth-trl
- https://ollama.com/library/llama3.2
- https://anythingllm.com/



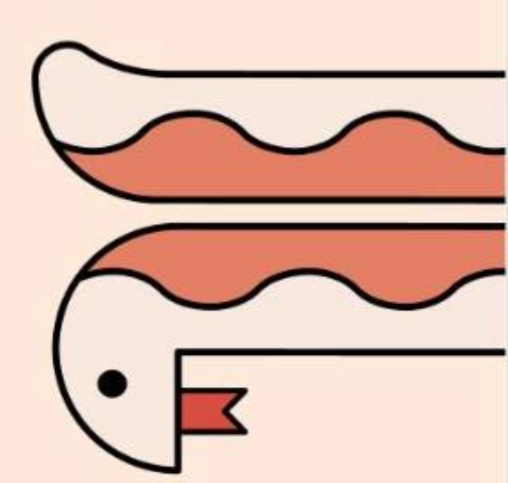
My github repo:

- https://github.com/Eleinad/talks_and_experiments/

ANNEX



Fine-Tuning vs RAG



Fine-Tuning

- Model “learns” new info, stores it internally
- No need for constantly updated data
- Reproduce style and tone of answers

RAG (Retrieval Augmented Generation)

- Model “looks up” info from external documents in real time
- Well-suited for scenarios needing up-to-date information
- Data is document-based

IT DEPENDS!