Xebia

USING PYTHON TO FINE-TUNE AND USE A LOCAL LLM IN AN APPLICATION

Eric Greene
eric@training4programmers.com

Goals for this Session

- What are Language Models?
- What is Fine-tuning?
- Fine-tune and Run LMs Locally
- Third-Party Services

What are Language Models?

- A **language model** is an AI system that learns patterns in language to predict and generate text
- **Large Language Models (LLMs)** are trained on vast datasets with billions of parameters for advanced, nuanced language understanding
- **Small Language Models (SLMs)** utilize fewer parameters and data, making it lightweight and efficient for simpler tasks

LLM or SLM?

- There’s no strict cutoff; distinctions are based on parameters, training data, and resource needs
- LLMs usually have billions of parameters, while SLMs are more lightweight
- Deployment context doesn’t change a model’s inherent classification
- A downsized version of an LLM (e.g., Llama) remains an LLM despite fewer parameters

Fine-Tuning

- Fine-tuning customizes a pre-trained model for specific tasks or domains
- It uses a smaller, task-specific dataset to adjust the model's parameters
- This process enhances performance without needing to train from scratch
- Fine-tuning is computationally efficient and cost-effective
- Careful calibration is needed to prevent overfitting or catastrophic forgetting

Fine-tune and Run LMs Locally

- There are a variety of ways to fine-tune and run LLMs locally
- A person can opt for libraries such as PyTorch, TensorFlow, or Hugging Face Transformers
- Alternatively, one can use third-party services such as OpenAI, Cohere, or Anthropic
- A middle road is to use higher-level libraries and third-party services to fine-tune a model and then run it locally
- This approach provides a balance between control and ease of use

Third-Party Services

- Hugging Face - huggingface.co - Supplies pre-trained models and fine-tuning utilities as the core framework for adapting language models
- Unsloth - unsloth.ai - Streamlines and optimizes the fine-tuning workflow—automating aspects of training to reduce complexity and speed up the process

Note: These were selected by the presenter and are for educational purposes only and do not represent an endorsement by HPE.

Third-Party Services

- Weight & Biases - wandb.com - Offers experiment tracking and visualization, allowing you to monitor metrics, compare runs, and fine-tune hyperparameters effectively
- Ollama - ollama.com - Serves as the deployment platform, enabling you to export and run your fine-tuned model locally for inference

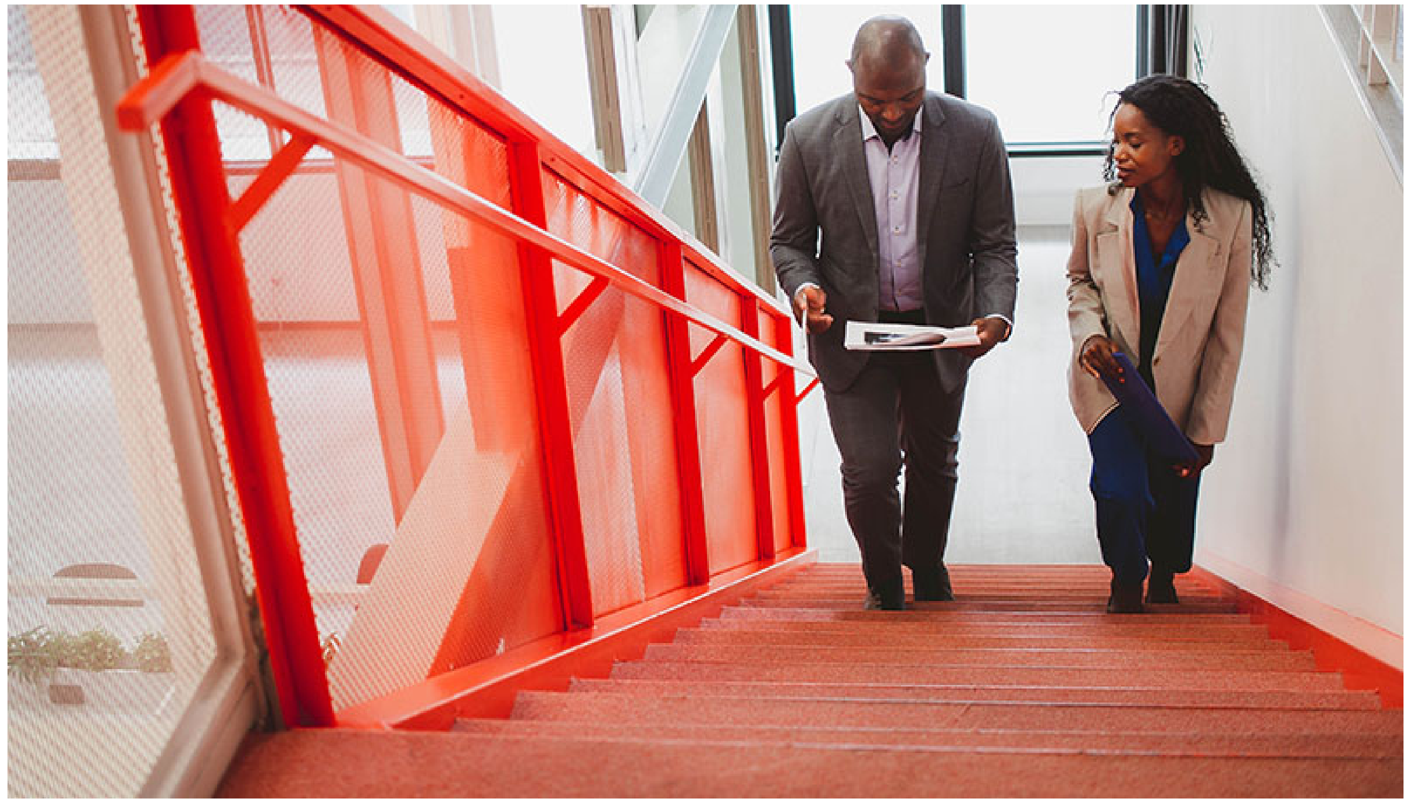
Note: These were selected by the presenter and are for educational purposes only and do not represent an endorsement by HPE.

Fine-tune and use a local LLM Demo

Let's Explore Fine-tuning a local LLM!

Fine-tune and use a local LLM Next Steps

- Explore pre-trained models and datasets on Hugging Face.
- Explore Unsloth and Weights & Biases to fine-tune models.
- Use Ollama to run the fine-tuned models locally.
- Review your personal and business datasets and applications to determine where they can be enhanced with LLMs.
- Run the code from the webinar and explore it on your own.
- Incorporate it into your next project!



Download the Code

github.com/cc-xebia-webinars/language-models_03112025

slides and source code available

Q&A

Questions?

Xebia

Thank you!

Eric Greene

eric@training4programmers.com