

Week 1 resources

Below you'll find links to the research papers discussed in this weeks videos. You don't need to understand all the technical details discussed in these papers - **you have already seen the most important points you'll need to answer the quizzes** in the lecture videos.

However, if you'd like to take a closer look at the original research, you can read the papers and articles via the links below.

1. **Generative AI Lifecycle**

* [**Generative AI on AWS: Building Context-Aware, Multimodal Reasoning Applications**](https://www.amazon.com/Generative-AI-AWS-Multimodal-Applications/dp/1098159225/) - This O'Reilly book dives deep into all phases of the generative AI lifecycle including model selection, fine-tuning, adapting, evaluation, deployment, and runtime optimizations.

1. **Transformer Architecture**

* [**Attention is All You Need**](https://arxiv.org/pdf/1706.03762) - This paper introduced the Transformer architecture, with the core “self-attention” mechanism. This article was the foundation for LLMs.
* [**BLOOM: BigScience 176B Model**](https://arxiv.org/abs/2211.05100) - BLOOM is a open-source LLM with 176B parameters trained in an open and transparent way. In this paper, the authors present a detailed discussion of the dataset and process used to train the model. You can also see a high-level overview of the model [here](https://bigscience.notion.site/BLOOM-BigScience-176B-Model-ad073ca07cdf479398d5f95d88e218c4).
* [**Vector Space Models**](https://www.coursera.org/learn/classification-vector-spaces-in-nlp/home/week/3) - Series of lessons from DeepLearning.AI's Natural Language Processing specialization discussing the basics of vector space models and their use in language modeling.

1. **Pre-training and scaling laws**

* [**Scaling Laws for Neural Language Models**](https://arxiv.org/abs/2001.08361)- empirical study by researchers at OpenAI exploring the scaling laws for large language models.

1. **Model architectures and pre-training objectives**

* [**What Language Model Architecture and Pretraining Objective Work Best for Zero-Shot Generalization?**](https://arxiv.org/pdf/2204.05832.pdf) - The paper examines modeling choices in large pre-trained language models and identifies the optimal approach for zero-shot generalization.
* [**HuggingFace Tasks**](https://huggingface.co/tasks) **and** [**Model Hub**](https://huggingface.co/models) - Collection of resources to tackle varying machine learning tasks using the HuggingFace library.
* [**LLaMA: Open and Efficient Foundation Language Models**](https://arxiv.org/pdf/2302.13971.pdf) - Article from Meta AI proposing Efficient LLMs (their model with 13B parameters outperform GPT3 with 175B parameters on most benchmarks)

1. **Scaling laws and compute-optimal models**

* [**Language Models are Few-Shot Learners**](https://arxiv.org/pdf/2005.14165.pdf)-This paper investigates the potential of few-shot learning in Large Language Models.
* [**Training Compute-Optimal Large Language Models**](https://arxiv.org/pdf/2203.15556.pdf)- Study from DeepMind to evaluate the optimal model size and number of tokens for training LLMs. Also known as “Chinchilla Paper”.
* [**BloombergGPT: A Large Language Model for Finance**](https://arxiv.org/pdf/2303.17564.pdf) - LLM trained specifically for the finance domain, a good example that tried to follow chinchilla laws.

Week 2 Resources

Below you'll find links to the research papers discussed in this weeks videos. You don't need to understand all the technical details discussed in these papers - **you have already seen the most important points you'll need to answer the quizzes** in the lecture videos.

However, if you'd like to take a closer look at the original research, you can read the papers and articles via the links below.

1. **Generative AI Lifecycle**

* [**Generative AI on AWS: Building Context-Aware, Multimodal Reasoning Applications**](https://www.amazon.com/Generative-AI-AWS-Multimodal-Applications/dp/1098159225/) - This O'Reilly book dives deep into all phases of the generative AI lifecycle including model selection, fine-tuning, adapting, evaluation, deployment, and runtime optimizations.

1. **Multi-task, instruction fine-tuning**

* [**Scaling Instruction-Finetuned Language Models**](https://arxiv.org/pdf/2210.11416.pdf) - Scaling fine-tuning with a focus on task, model size and chain-of-thought data.
* [**Introducing FLAN: More generalizable Language Models with Instruction Fine-Tuning**](https://ai.googleblog.com/2021/10/introducing-flan-more-generalizable.html) - This blog (and article) explores instruction fine-tuning, which aims to make language models better at performing NLP tasks with zero-shot inference.

1. **Model Evaluation Metrics**

* [**HELM - Holistic Evaluation of Language Models**](https://crfm.stanford.edu/helm/latest/) - HELM is a living benchmark to evaluate Language Models more transparently.
* [**General Language Understanding Evaluation (GLUE) benchmark**](https://openreview.net/pdf?id=rJ4km2R5t7) - This paper introduces GLUE, a benchmark for evaluating models on diverse natural language understanding (NLU) tasks and emphasizing the importance of improved general NLU systems.
* [**SuperGLUE**](https://super.gluebenchmark.com/) - This paper introduces SuperGLUE, a benchmark designed to evaluate the performance of various NLP models on a range of challenging language understanding tasks.
* [**ROUGE: A Package for Automatic Evaluation of Summaries**](https://aclanthology.org/W04-1013.pdf) - This paper introduces and evaluates four different measures (ROUGE-N, ROUGE-L, ROUGE-W, and ROUGE-S) in the ROUGE summarization evaluation package, which assess the quality of summaries by comparing them to ideal human-generated summaries.
* [**Measuring Massive Multitask Language Understanding (MMLU)**](https://arxiv.org/pdf/2009.03300.pdf) - This paper presents a new test to measure multitask accuracy in text models, highlighting the need for substantial improvements in achieving expert-level accuracy and addressing lopsided performance and low accuracy on socially important subjects.
* [**BigBench-Hard - Beyond the Imitation Game: Quantifying and Extrapolating the Capabilities of Language Models**](https://arxiv.org/pdf/2206.04615.pdf) - The paper introduces BIG-bench, a benchmark for evaluating language models on challenging tasks, providing insights on scale, calibration, and social bias.

1. **Parameter- efficient fine tuning (PEFT)**

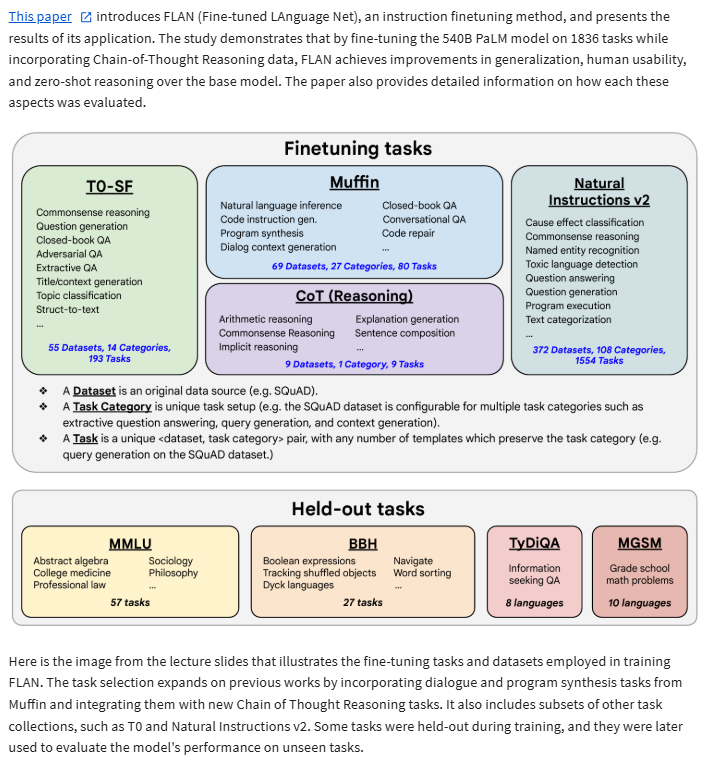
* [**Scaling Down to Scale Up: A Guide to Parameter-Efficient Fine-Tuning**](https://arxiv.org/pdf/2303.15647.pdf) - This paper provides a systematic overview of Parameter-Efficient Fine-tuning (PEFT) Methods in all three categories discussed in the lecture videos.
* [**On the Effectiveness of Parameter-Efficient Fine-Tuning**](https://arxiv.org/pdf/2211.15583.pdf) - The paper analyzes sparse fine-tuning methods for pre-trained models in NLP.

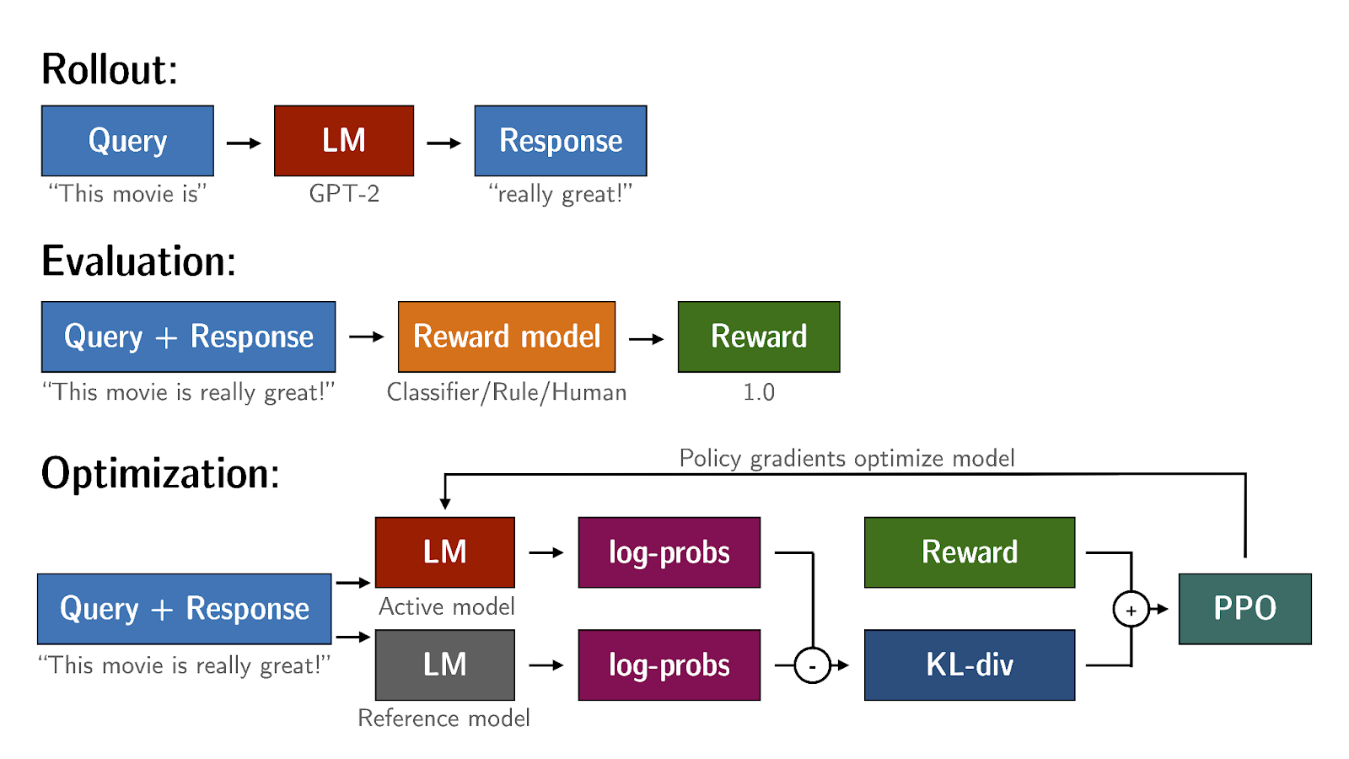
1. **LoRA**

* [**LoRA Low-Rank Adaptation of Large Language Models**](https://arxiv.org/pdf/2106.09685.pdf)- This paper proposes a parameter-efficient fine-tuning method that makes use of low-rank decomposition matrices to reduce the number of trainable parameters needed for fine-tuning language models.
* [**QLoRA: Efficient Finetuning of Quantized LLMs**](https://arxiv.org/pdf/2305.14314.pdf) - This paper introduces an efficient method for fine-tuning large language models on a single GPU, based on quantization, achieving impressive results on benchmark tests.

1. **Prompt tuning with soft prompts**

* [**The Power of Scale for Parameter-Efficient Prompt Tuning**](https://arxiv.org/pdf/2104.08691.pdf) - The paper explores "prompt tuning," a method for conditioning language models with learned soft prompts, achieving competitive performance compared to full fine-tuning and enabling model reuse for many tasks.



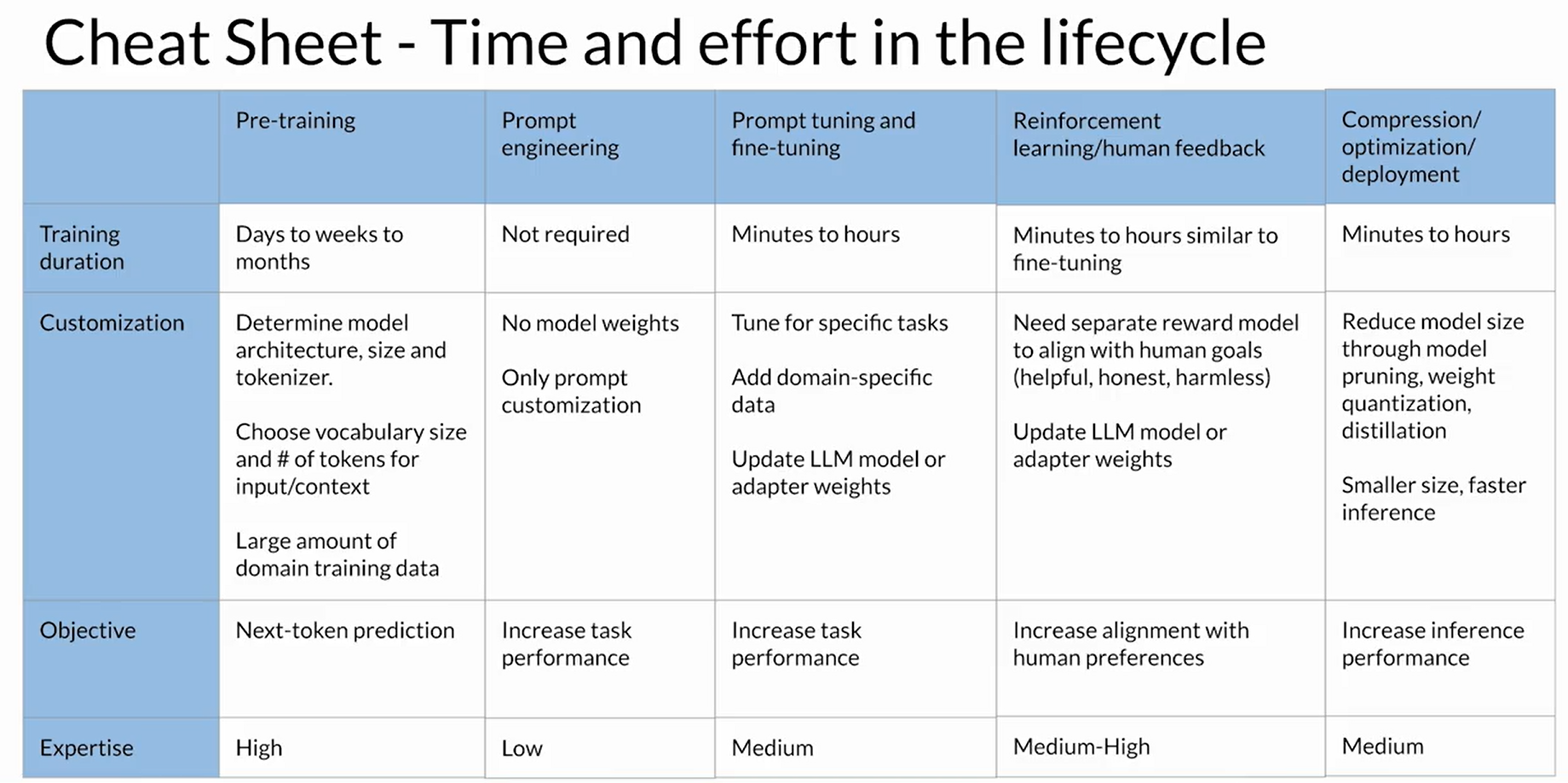


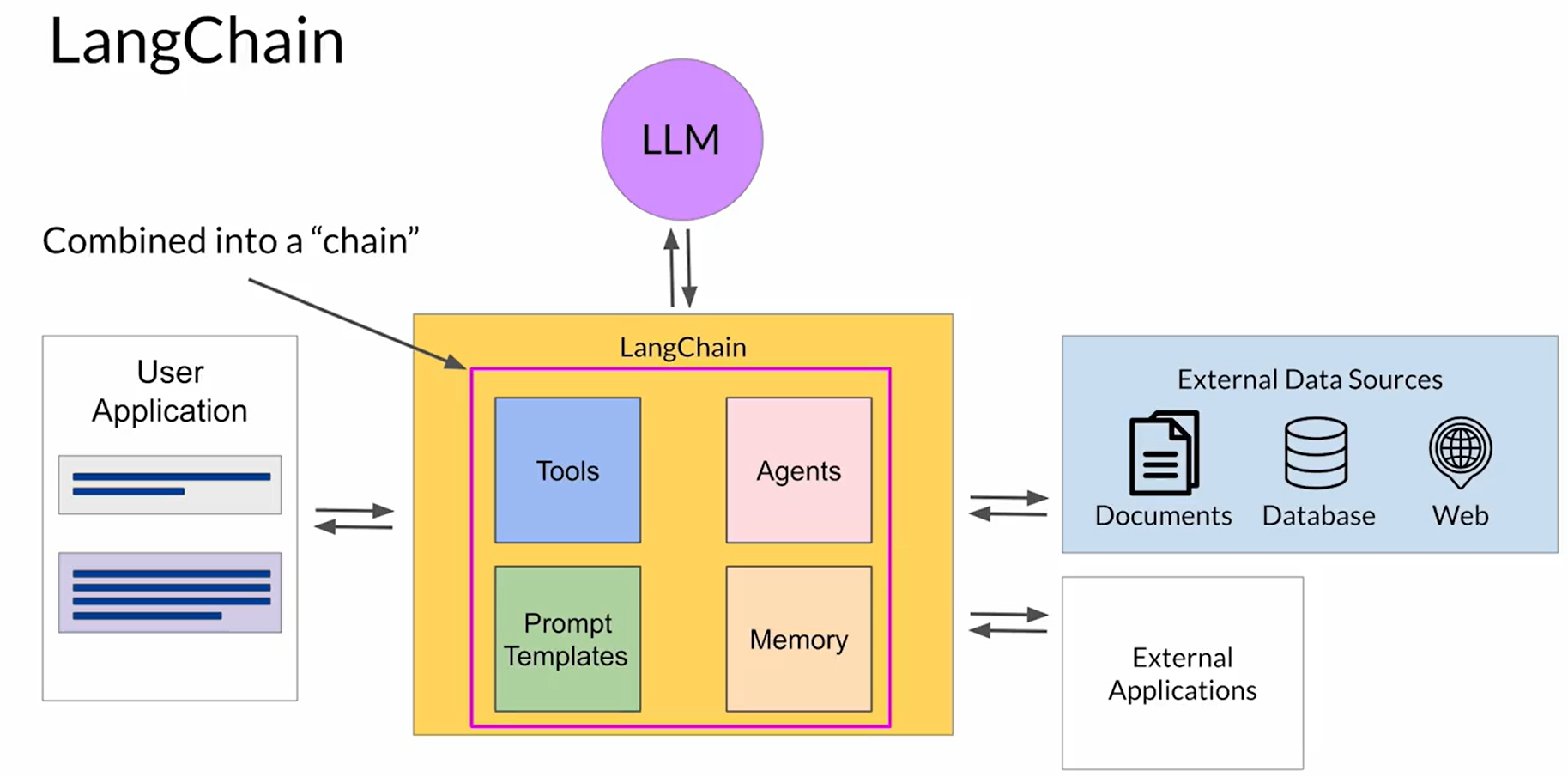
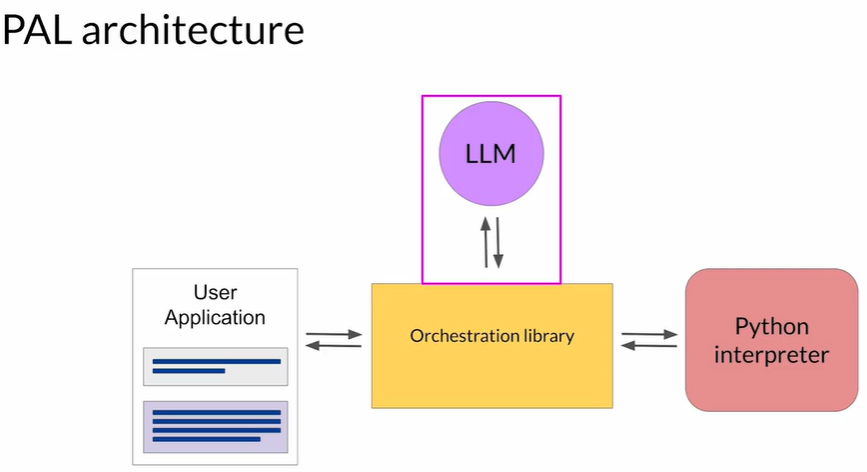
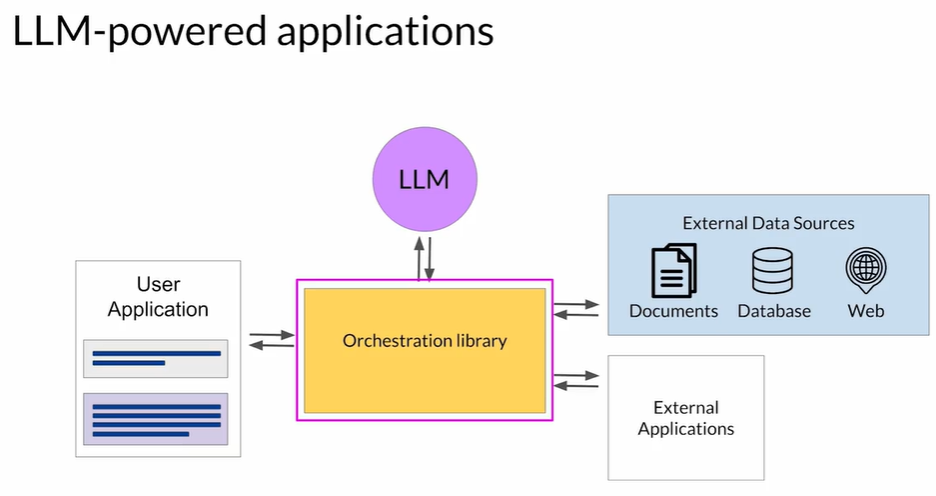
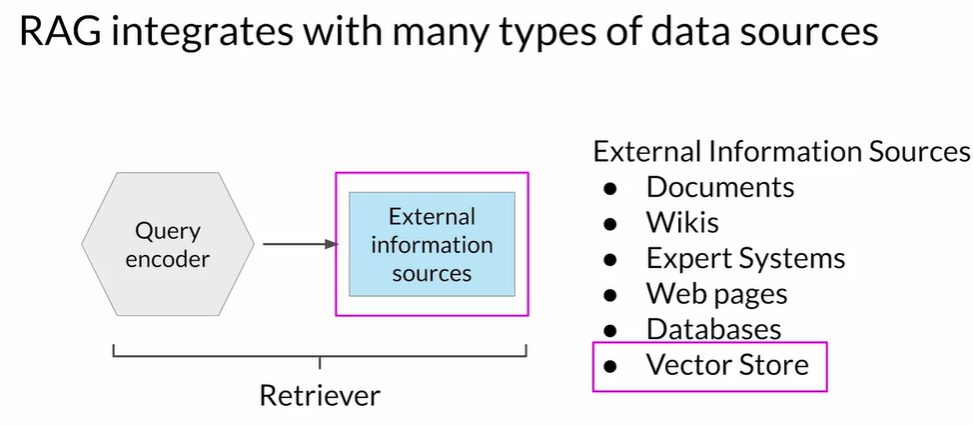
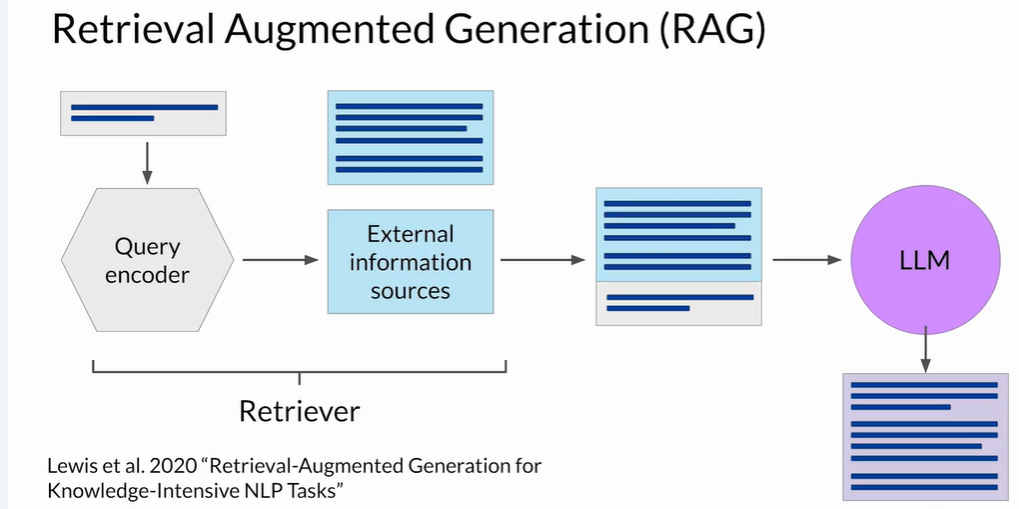
KL-Divergence, or Kullback-Leibler Divergence, is a concept often encountered in the field of reinforcement learning, particularly when using the Proximal Policy Optimization (PPO) algorithm. It is a mathematical measure of the difference between two probability distributions, which helps us understand how one distribution differs from another. In the context of PPO, KL-Divergence plays a crucial role in guiding the optimization process to ensure that the updated policy does not deviate too much from the original policy.

In PPO, the goal is to find an improved policy for an agent by iteratively updating its parameters based on the rewards received from interacting with the environment. However, updating the policy too aggressively can lead to unstable learning or drastic policy changes. To address this, PPO introduces a constraint that limits the extent of policy updates. This constraint is enforced by using KL-Divergence.

To understand how KL-Divergence works, imagine we have two probability distributions: the distribution of the original LLM, and a new proposed distribution of an RL-updated LLM. KL-Divergence measures the average amount of information gained when we use the original policy to encode samples from the new proposed policy. By minimizing the KL-Divergence between the two distributions, PPO ensures that the updated policy stays close to the original policy, preventing drastic changes that may negatively impact the learning process.

A library that you can use to train transformer language models with reinforcement learning, using techniques such as PPO, is TRL (Transformer Reinforcement Learning). In [this link](https://huggingface.co/blog/trl-peft) you can read more about this library, and its integration with PEFT (Parameter-Efficient Fine-Tuning) methods, such as LoRA (Low-Rank Adaption). The image shows an overview of the PPO training setup in TRL.

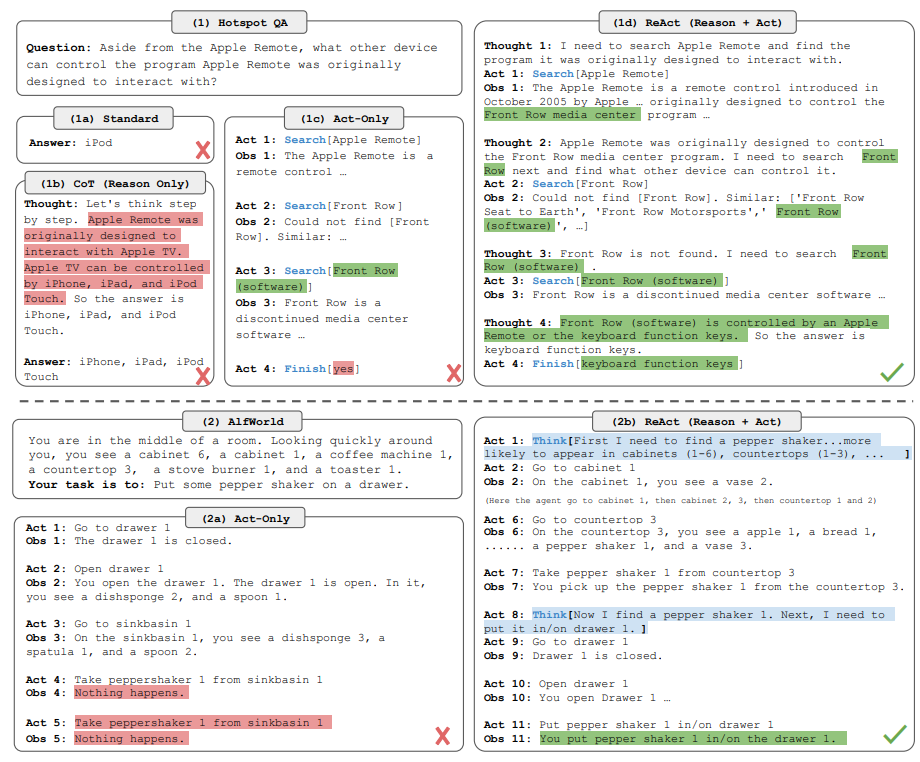




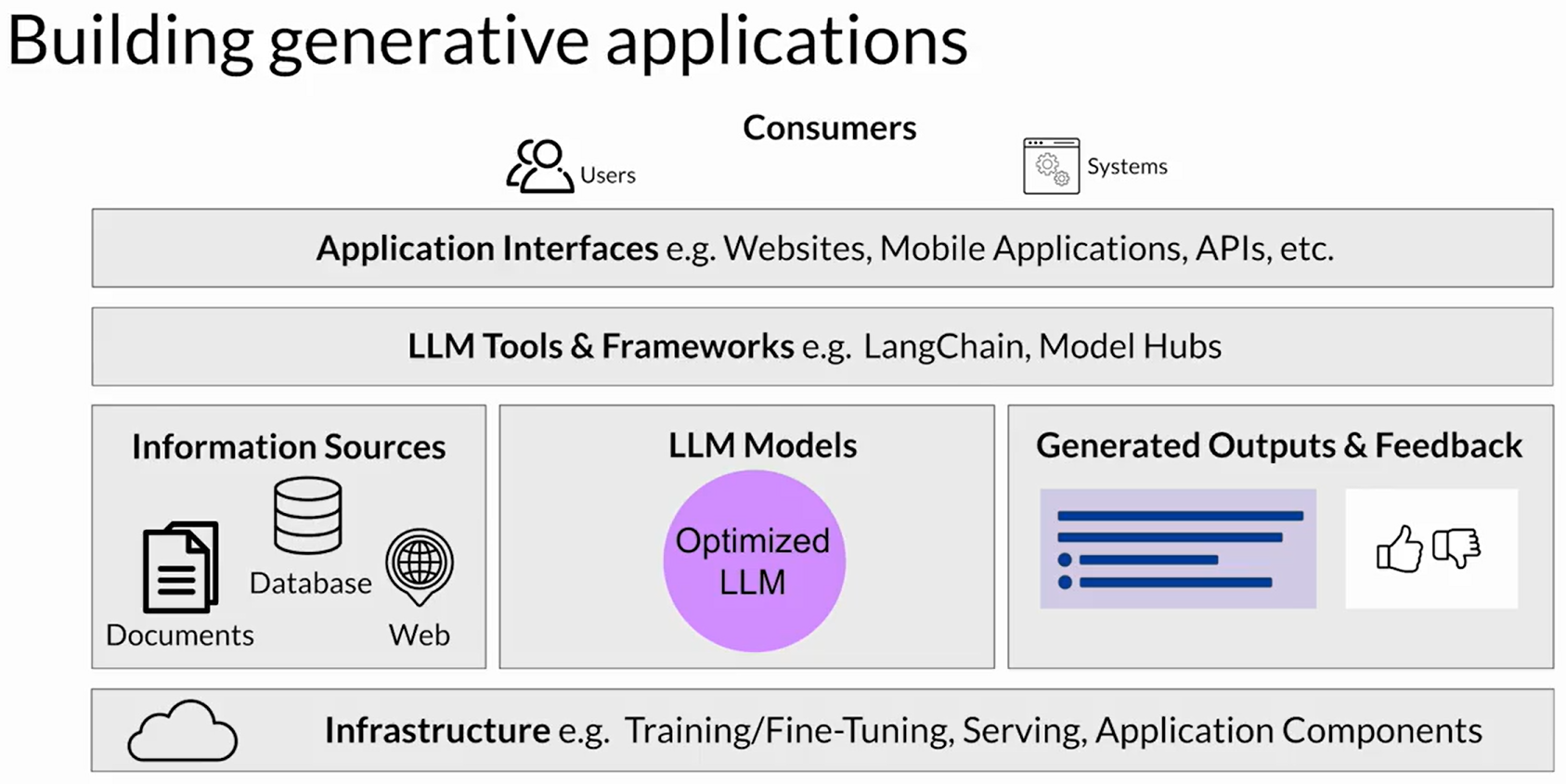
ReAct: Reasoning and action

[This paper](https://arxiv.org/abs/2210.03629) introduces ReAct, a novel approach that integrates verbal reasoning and interactive decision making in large language models (LLMs). While LLMs have excelled in language understanding and decision making, the combination of reasoning and acting has been neglected. ReAct enables LLMs to generate reasoning traces and task-specific actions, leveraging the synergy between them. The approach demonstrates superior performance over baselines in various tasks, overcoming issues like hallucination and error propagation. ReAct outperforms imitation and reinforcement learning methods in interactive decision making, even with minimal context examples. It not only enhances performance but also improves interpretability, trustworthiness, and diagnosability by allowing humans to distinguish between internal knowledge and external information.

In summary, ReAct bridges the gap between reasoning and acting in LLMs, yielding remarkable results across language reasoning and decision making tasks. By interleaving reasoning traces and actions, ReAct overcomes limitations and outperforms baselines, not only enhancing model performance but also providing interpretability and trustworthiness, empowering users to understand the model's decision-making process.



**Image:** The figure provides a comprehensive visual comparison of different prompting methods in two distinct domains. The first part of the figure (1a) presents a comparison of four prompting methods: Standard, Chain-of-thought (CoT, Reason Only), Act-only, and ReAct (Reason+Act) for solving a HotpotQA question. Each method's approach is demonstrated through task-solving trajectories generated by the model (Act, Thought) and the environment (Obs). The second part of the figure (1b) focuses on a comparison between Act-only and ReAct prompting methods to solve an AlfWorld game. In both domains, in-context examples are omitted from the prompt, highlighting the generated trajectories as a result of the model's actions and thoughts and the observations made in the environment. This visual representation enables a clear understanding of the differences and advantages offered by the ReAct paradigm compared to other prompting methods in diverse task-solving scenarios.



Week 3 resources

Below you'll find links to the research papers discussed in this weeks videos. You don't need to understand all the technical details discussed in these papers - **you have already seen the most important points you'll need to answer the quizzes** in the lecture videos.

However, if you'd like to take a closer look at the original research, you can read the papers and articles via the links below.

1. **Generative AI Lifecycle**

* [**Generative AI on AWS: Building Context-Aware, Multimodal Reasoning Applications**](https://www.amazon.com/Generative-AI-AWS-Multimodal-Applications/dp/1098159225/) - This O'Reilly book dives deep into all phases of the generative AI lifecycle including model selection, fine-tuning, adapting, evaluation, deployment, and runtime optimizations.

1. **Reinforcement Learning from Human-Feedback (RLHF)**

* [**Training language models to follow instructions with human feedback**](https://arxiv.org/pdf/2203.02155.pdf) **-** Paper by OpenAI introducing a human-in-the-loop process to create a model that is better at following instructions (InstructGPT).
* [**Learning to summarize from human feedback**](https://arxiv.org/pdf/2009.01325.pdf) - This paper presents a method for improving language model-generated summaries using a reward-based approach, surpassing human reference summaries.

1. **Proximal Policy Optimization (PPO)**

* [**Proximal Policy Optimization Algorithms**](https://arxiv.org/pdf/1707.06347.pdf) - The paper from researchers at OpenAI that first proposed the PPO algorithm. The paper discusses the performance of the algorithm on a number of benchmark tasks including robotic locomotion and game play.
* [**Direct Preference Optimization: Your Language Model is Secretly a Reward Model**](https://arxiv.org/pdf/2305.18290.pdf) - This paper presents a simpler and effective method for precise control of large-scale unsupervised language models by aligning them with human preferences.

1. **Scaling human feedback**

* [**Constitutional AI: Harmlessness from AI Feedback**](https://arxiv.org/pdf/2212.08073.pdf)-This paper introduces a method for training a harmless AI assistant without human labels, allowing better control of AI behavior with minimal human input.

1. **Advanced Prompting Techniques**

* [**Chain-of-thought Prompting Elicits Reasoning in Large Language Models**](https://arxiv.org/pdf/2201.11903.pdf) - Paper by researchers at Google exploring how chain-of-thought prompting improves the ability of LLMs to perform complex reasoning.
* [**PAL: Program-aided Language Models**](https://arxiv.org/abs/2211.10435) - This paper proposes an approach that uses the LLM to read natural language problems and generate programs as the intermediate reasoning steps.
* [**ReAct: Synergizing Reasoning and Acting in Language Models**](https://arxiv.org/abs/2210.03629) This paper presents an advanced prompting technique that allows an LLM to make decisions about how to interact with external applications.

1. **LLM powered application architectures**

* [**LangChain Library (GitHub)**](https://github.com/hwchase17/langchain)- This library is aimed at assisting in the development of those types of applications, such as Question Answering, Chatbots and other Agents. You can read the documentation [here](https://docs.langchain.com/docs/).
* [**Who Owns the Generative AI Platform?**](https://a16z.com/2023/01/19/who-owns-the-generative-ai-platform/)- The article examines the market dynamics and business models of generative AI.