LLM Fundamentals Workshop: One-Page Overview

Workshop Title:
Understanding Large Language Models: From Pre-training to Production

Duration: 3-4 hours (including breaks)
Audience: Developers, data scientists, product managers, technical decision-makers
Focus: Demystifying LLMs through the complete training pipeline and practical usage patterns

---

Workshop Structure & Timing

Part	Topic	Duration
1	What Are LLMs? The Mental Model	15 min
2	Stage 1: Pre-training - Building the Base Model	30 min
3	Tokenization Deep Dive	20 min
BREAK		10 min
4	Stage 2: Supervised Fine-Tuning (SFT) - Creating Assistants	30 min
5	LLM Psychology: Hallucinations, Tools & Sharp Edges	35 min
BREAK		10 min
6	Stage 3: Reinforcement Learning - Teaching Models to Think	30 min
7	RLHF vs RL in Verifiable Domains	20 min
8	Practical Usage, Resources & Q&A	20 min

Key Learning Objectives

- Understand the three-stage training pipeline: Pre-training -> SFT -> RL
- Develop accurate mental models for what LLMs are and aren't
- Learn why models hallucinate and how to mitigate it
- Understand the difference between base models, assistant models, and reasoning models

- Recognize cognitive limitations and sharp edges in LLM capabilities
- Make informed decisions about model selection and usage patterns
- Understand the economics and compute requirements of training LLMs

---

Core Concepts Covered

Stage 1: Pre-training (Internet Document Simulator)

- Downloading and filtering internet data (Common Crawl -> Fine Web dataset)
- Tokenization: Converting text into token sequences (BPE algorithm, ~100K vocabulary)
- Neural network training: Predicting next token in sequence
- Base models as "lossy compression" of the internet
- Compute requirements: GPUs, data centers, costs ($40K -> $600 for GPT-2 reproduction)

Stage 2: Supervised Fine-Tuning (Creating Assistants)

- Conversation datasets: Human-labeled ideal responses
- Labeling instructions and data curation process
- Training on multi-turn conversations (not internet documents)
- What you're really talking to: Statistical simulation of human labelers
- Modern approach: Synthetic data + human editing

Stage 3: Reinforcement Learning (Discovering Thinking Strategies)

- Verifiable domains: Math, code (correct answers exist)
- Trial-and-error learning across thousands of solutions
- Emergent reasoning: Chain-of-thought, self-correction, backtracking
- Comparison to AlphaGo and "Move 37" moments
- Reasoning models vs assistant models (O3, DeepSeek-R1)

RLHF vs True RL:

- Unverifiable domains: Creative writing, summarization
- Reward model training on human preferences
- Limitations: Adversarial examples, gaming the reward model
- Why RLHF is "fine-tuning" not "magic RL"

LLM Psychology & Sharp Edges

Knowledge Architecture:

- Parameters = vague recollection (long-term memory)

- Context window = working memory (direct access)
- Always copy-paste relevant content into context

Common Failure Modes:

- Hallucinations: Making up facts confidently
- Counting failures: "How many Rs in strawberry?"
- Spelling tasks: Token-level vs character-level processing
- Unexpected failures: "9.11 vs 9.9" (Bible verse neurons)

Mitigation Strategies:

- Tool use: Web search, code interpreter
- Interrogation-based training for factuality
- Distributing computation across tokens
- Using structured outputs and verification

Interactive Elements

- Live tokenization demo (TikTokenizer visualization)
- Base model vs assistant model comparison
- Reasoning model thought process exploration
- Q&A on specific use cases and model selection

Key Technical Insights

The Three Training Stages = How We Train Children:

1. Pre-training = Reading textbook exposition (knowledge acquisition)
2. SFT = Studying worked examples (imitating experts)
3. RL = Practice problems (discovering strategies)

Critical Differences:

- Base models: Token autocomplete, internet simulator
- Assistant models: Simulate human labelers following instructions
- Reasoning models: Emergent thinking strategies from RL

Tokens Are Everything:

- Models process sequences left-to-right, one token at a time
- Finite compute per token (~100 layers)

- Must distribute reasoning across many tokens
- "Models need tokens to think"

Practical Takeaways

Model Selection Guide:

- GPT-4o, Claude Sonnet: General assistant tasks (80-90% of use cases)
- O3, DeepSeek-R1: Complex reasoning, math, code problems
- Base models: Rare, mostly for research or specific fine-tuning

Best Practices:

- Never fully trust LLM outputs—always verify
- Use tools (web search, code) over mental arithmetic
- Paste context directly rather than relying on memory
- Understand token limits and context management
- Test multiple models for critical applications

Where to Access Models:

- ChatGPT (chat.openai.com) - proprietary
- Claude (claude.ai) - proprietary
- Gemini (gemini.google.com) - proprietary
- DeepSeek (chat.deepseek.com) - open weights
- Together.ai, Hyperbolic - inference providers
- LM Studio - local model hosting

Resources for Further Learning

Stay Updated:

- LMSys Arena (lmarena.ai) - Model leaderboard with human rankings
- AI News Newsletter - Comprehensive daily updates
- X/Twitter - Follow AI researchers and practitioners

Key Papers:

- InstructGPT (OpenAI, 2022) - SFT methodology
- DeepSeek-R1 (2025) - RL for reasoning models
- GPT-2 Paper (2019) - First modern LLM architecture

Hands-on Resources:

- TikTokenizer - Token visualization
- Transformer visualization - Neural network internals
- LM Studio - Local model experimentation

Main Source: Andrej Karpathy - Deep Dive into LLMs like ChatGPT

- https://newsletter.systemdesign.one/p/llm-concepts

Contact & Preparation

Presenter: Mehdi Maleki
Email: mosioc79@gmail.com
Pre-workshop: No prerequisites required, but familiarity with basic ML concepts helpful. Bring questions about specific LLM use cases!