AI & LLM 快速上手指南

Topic	Lectures	Supplements	Exercises
ML Foundations			
Math Foundations	Linear Algebra Review Probability Review Differential Calculus Review		
Supervised Learning	快速了解機器學習基本原理 (1/2) 快速了解機器學習基本原理 (2/2) Regression Where does the error come from? Gradient Descent Classification Logistic Regression	CS229 - Supervised Learning Note CS229 - Generative Algorithms	
Neural Networks and Backpropagation	Brief Introduction of Deep Learning Backpropagation	CS231n - Backprop Note CS224n - Computing Neural Network Gradients	CS231n - Backprop Tutorial
Training Neural Networks	General Guidance When Gradient Is Small: Local Minimum and Saddle Point Tips for Training: Batch and Momentum Tips for Training: Adaptive Learning Rate Tips for Training: Batch Normalization	Lilian Weng - An Overview of Deep Learning for Curious People Andrej Karpathy - A Recipe for Training Neural Networks CS231n - Neural Networks Note 1 CS231n - Neural Networks Note 2 CS231n - Neural Networks Note 3	

PyTorch Tutorial	Tutorial 2: Introduction to PyTorch Tutorial 3: Activation Functions Tutorial 4: Optimization and Initialization	Guide 2: Research Project with PyTorch Guide 3: Debugging in PyTorch HDL - Introduction to HyperParameter Tuning HDL - Introduction to Multi GPU Programming CS197 - Organizing Model Training with Weights & Biases and Hydra	CS231n - Assignment 2 Tensor Puzzles
Attention and Transformers			
Attention	Self-Attention (1/2) Self-Attention (2/2)	CS224n - Language Models, RNN, GRU and LSTM	
Transformers	CS224n - Self-Attention and Transformers Transformer (1/2) Transformer (2/2)	The Illustrated Transformer CS324 - LLM Introduction	Tutorial 6: Transformers and Multi-Head Attention Tutorial 15: Vision Transformers
Pretraining	CS224n - Pretraining		
HuggingFace Transformers Tutorial	CS224n - HuggingFace Transformers Tutorial	HuggingFace Transformers Course HuggingFace Transformers Doc	
Generative AI			
LLM Basics	Andrej Karpathy - Intro to Large Language Models Andrej Karpathy - State of GPT ChatGPT (可能)是怎麼煉成的 ChatGPT 原理剖析 (1/3) — 對 ChatGPT 的常見誤解 ChatGPT 原理剖析 (2/3) — 預訓練 (Pre-train) ChatGPT 原理剖析 (3/3) — ChatGPT 所帶來的研究問題	CS324 LLM Capabilities CS324 LLM Modeling Building Systems with the ChatGPT API OpenAI Cookbook	

Prompting	Finetuning vs. Prompting：對於大型語言模型的不同期待所衍生的兩類使用方式 (1/3) Finetuning vs. Prompting：對於大型語言模型的不同期待所衍生的兩類使用方式 (2/3) Finetuning vs. Prompting：對於大型語言模型的不同期待所衍生的兩類使用方式 (3/3)	Lilian Weng - Prompt Engineering OpenAI Prompt Engineering ChatGPT Prompt Engineering for Developers	CS324 Assignment
RL Basics (Optional)	DRL: Basics DRL: Policy Gradient DRL: Actor-Critic DRL: When Reward Is Sparse DRL: Imitation Learning	OpenAI Spinning Up 动手学强化学习 Lilian Weng - A (Long) Peek into Reinforcement Learning Pieter Abbeel - Foundations of Deep RL HuggingFace Deep RL Course	
RLHF	CS224n - Prompting, RLHF OpenAI - Aligning language models to follow instructions	Chip Huyen - RLHF: Reinforcement Learning from Human Feedback	
Diffusion Model	生成式學習的兩種策略：要各個擊破，還是要一次到位 速覽圖像生成常見模型 淺談圖像生成模型 Diffusion Model 原理 Stable Diffusion、DALL-E、Imagen 背後共同的套路 Diffusion Model 原理剖析 (1/4) Diffusion Model 原理剖析 (2/4) Diffusion Model 原理剖析 (3/4) Diffusion Model 原理剖析 (4/4)	Lilian Weng - What are Diffusin Models? The Illustrated Stable Diffusion	
LLM	大模型 + 大資料 = 神奇結果? (1/3)：大模型的頓悟時刻 大模型 + 大資料 = 神奇結果? (2/3)：到底要多少資料才夠 GPT-4 來了! GPT-4 這次有什麼神奇的能力呢?	COS597G - Understanding Large Language Models The Technology Behind BLOOM Training Andrej Karpathy - Let's build GPT: from scratch, in code,	

	窮人如何低資源復刻自己的 ChatGPT ChatGPT 可以自我反省! 讓 AI 村民組成虛擬村莊會發生甚麼事? CS324 LLM Training CS324 LLM Adaptation	spelled out. Chip Huyen - Building LLM applications for production Lilian Weng - LLM Powered Autonomous Agents Chip Huyen - Open challenges in LLM research	

Notes:

1. This onboarding tutorial is designed to equip new interns with the foundational knowledge necessary for engaging in GenAI & LLM projects.
2. Providing an exhaustive introduction to machine learning or deep learning is not the goal of this tutorial. For a more detailed study, please refer to the courses listed below.
3. The content should be self-sufficient. A basic understanding of linear algebra, calculus, probability, and statistics is assumed. However, if you lack these foundations, you can still begin and learn them when needed.
4. For beginners, it typically takes 1-2 months to complete this tutorial.
5. These materials have been curated and structured to cover specific topics. If you find some sections repetitive or too basic, you're welcome to skip them. Conversely, if you encounter topics with a steep learning curve, seek additional resources or supplemental materials for assistance.
6. Exercises are crucial for honing your coding skills.
7. Be prepared to encounter and overcome potential language barriers.

References:

Courses:

- https://speech.ee.ntu.edu.tw/~hylee/ml/2023-spring.php
- https://speech.ee.ntu.edu.tw/~hylee/ml/2021-spring.php
- http://cs231n.stanford.edu/schedule.html
- https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/schedule.html
- https://stanford-cs324.github.io/winter2022/

- https://web.stanford.edu/class/cs224n/
- https://www.deeplearning.ai/short-courses/
- https://cs229.stanford.edu/syllabus-fall2021.html

Blogs & Tutorials:

- https://lilianweng.github.io/posts/
- https://openai.com/research
- https://karpathy.github.io/
- https://huyenchip.com/blog/