YUYANG ZHANG

✉ yy.zhang.2023@mitb.smu.edu.sg ☎ +65 86881560 https://www.linkedin.com/in/zhang-yuyang/

https://github.com/Yyzhang2000?tab=repositories https://yyzhang2000.github.io/Blog/

EDUCATION

SINGAPORE MANAGEMENT UNIVERSITY **Aug 2023 - Mar 2025**
Master of IT in Business (AI Track)
(Applied Machine Learning: A, Introduction to AI: A+, Blockchain Technology: A-, Learning and Planning in Intelligent Systems: A+)

University of Wollongong **Mar 2020 – Mar 2023**
Bachelor of Computer Science (Major in Big Data)

- Distinction

Internship

SMART Research Assistant – Generative Model **Mar-2025-Cur**

- Preprocessed and curated datasets for training generative models, including resizing, normalization and data augmentation
- Evaluated model performance using metrics such as FID, contextual loss, and Style Loss
- Collaborated with the research team to benchmark different architectures and experiment with various training strategies.

ACADEMIC PROJECTS

Stable Diffusion with Control Net [Blog][Code] **Mar-2025**

- Implemented Stable Diffusion from scratch using Pytorch
- Used CLIP as the text encoder and a Variational AutoEncoder to compress images into latent space.
- Built and trained a U-Net architecture to model the noise prediction and generated high-quality images.
- Applied Class-Free Guidance to enhance image generation quality and text-image relevance.
- Trained the diffusion model on the MNIST dataset to validate implementation correctness and understand training dynamics
- Loaded weighted from pre-trained Stable Diffusion V1-5 model and file-tuned it uses ControlNet on the COCO Caption enable better conditional generation based on edge maps and other input conditions.

ChatGPT-2 **Nov-2024**

- Building GPT-2 model(124M) using PyTorch from scratch,
- Using the FineWeb-10B datasets(25GB) trained on the A100 GPU.
- Using the Distributed Data Parallel to speed up the training process.
- Also implement Weight Decay, Gradient Accumulation, Learning Rate warm up and decay to stable the training process.
- After training the GPT-2 model. Fine-tuning the model using Supervised Fine-Tuning method, to give the model instruction following ability, and after that, trained the model using RLHF using PPO and DPO algorithms.
- What I have learned from this project is how to train, fine-tuning the LLM. This project let me become more familiar with the process of training and fine-tuning LLM.

PaliGemma

- This is the multi-modal large language Model,
- I built the model from scratch using the PyTorch and load weight from the Huggingface
- In this project, I improve the model efficiency with advanced inference techniques.

Build LLaMa2 from scratch **Oct-2024**

- Build the Large Language Model using PyTorch from scratch, including Rotary Position embedding, KV cache, Multi-Query Attention, Grouped Query Attention
- Loaded the weights from the LLaMa2 website and fine tune the model with LoRA on the CoLab.

- Implement the Top-K decoding method.
- After load the weight, Implement the LoRA and QLoRA and adapter layer to fine-tune the model.

Deep Reinforcement Learning Pong **Oct-2024**

- This is project, I implement several classic DRL algorithm,
- Valued Based Methods: DQN, double DQN, dueling DQN
- Policy Gradient: REINFORCE
- Actor-Critic: SAC, and PPO
- After implementing those algorithms, compare with each other.

Deep Learning Framework **Apr-2024**

- This project is to get familiar with the basic components used in the Deep Learning, it includes several important concepts such as: Normalization, Dropout, Different Activation Function. The most important part is that it builds a tensor which enable auto-differential like PyTorch.
- Using the customized framework to implement the several basic and classic neural network architectures, for example, Convolution NN, Recurrent NN and Transformer Architecture.

Besides all above projects, during my free time, I like to write blogs. So far, I have written the following content:

- Large Language Model[Link]: the components in the LLM, and the techniques to post-training the LLM and evaluate the LLM
- Overview of Generative Model[Link]: introduce six deep generative models: Auto-Regressive models, VAE, Diffusion Models, Flow Models, GAN, Energy-Based Model.
- Reinforcement Learning[Link]: RL techniques from basic MDP to the advanced algorithms such as DQN, PPO.

ADDITIONAL

- Certifications: Deep Learning(Coursera), Mathematic For Machine Learning(Coursera), Machine Learning(Coursera)
- Technical Skills: Python, Machine Learning, Deep Learning, Deep Reinforcement Learning, Large Langue Model, Computer Vision, Natural Language Processing, Generative Models, Probabilistic Graphic Model, Bayesian Inference.
- Language Skills: English (Business Professional), Mandarin (Native)
- Work Authorization: Singapore (Part-time Internship), China (Citizen)