**MLE Class**

Scale Machine Learning System

3 hướng tiếp cận xây dựng hệ thống học máy lớn

* **Nguyên tắc 1**: Tối ưu dữ liệu
* **Nguyên tắc 2**: Sử dụng các mô hình/kỹ thuật tiên tiến
* **Nguyên tắc 3:** Sử dụng tối đa năng lực tính toán của phần cứng

|  |  |  |
| --- | --- | --- |
| **Buổi** | **Thời gian** | **Nội dung** |
|  |  | **Gặp gỡ đội ngũ giảng dạy**  Giới thiệu về lớp học, giảng viên, trao đổi với học viên về ứng dụng AI trong công việc |
|  |  | **Ôn tập Học máy. Các thuật toán training.**  Ôn tập học máy (Video | Quizz)  **Ôn tập các mô hình cơ bản**   * Logistic Regression * Softmax Regression * Mạng nơ ron * CNN * RNN * **Giới thiệu qua Transformer** * **Ứng dụng Transformer** |
|  |  | **Ôn tập lập trình mô hình**   * Lập trình bài toán phân loại hình ảnh + dự đoán âm thanh Tensorflow * Họ **Transformer** và các mô hình ứng dụng**: Bert/GPT**   **Giới thiệu Pytorch**   * Các hàm cơ bản * Xây dựng mô hình cơ bản với Pytorch * **Dataset và Dataloader** trong Pytorch * Custom Training với Pytorch   **Thư viện** **HuggingFace**   * **Tokenizer trong HuggingFace** * Ý tưởng BPE (Byte Pair Encoding) * Thư viện **Hugging Face** cho NLP * Bài toán phân loại trên Tiếng Việt   Notebook Pytorch | Phân loại văn bản tiếng Việt với HuggingFace |
|  |  | **[Live Coding] Làm việc và lưu trữ, theo dõi dữ liệu với MongoDB.**  **Hướng dẫn setup MongoDB**   * Đăng ký tài khoản MongoDB Atlas * Các cách kết nối MongoDB. Doc * Thư viện PyMongo * Liệt kê, thêm sửa, xoá dữ liệu   Thiết kế **Pipeline clean dữ liệu thô** và theo dõi sự thay đổi của dữ liệu  **Làm việc và kiểm tra chất lượng dữ liệu**   * **TDD (**Test-Driven Development**)** và quy trình review test/code. * Kiểm thử quá trình tiền xử lý dữ liệu * Thư viện [unittest](https://docs.python.org/3/library/unittest.html). |
|  |  | **Các kỹ thuật training hiệu quả 1 (Nguyên tắc 2)**   * Tối ưu hàm lồi (Convex Optimization) * Sử dụng các thuật toán training. AdaGrad/Adam * Cài đặt các tham số khởi tạo * Grid Search/Random Search |
|  |  | **[Live Coding] Phân bổ dữ liệu để training. Đặt giờ tự động các tác vụ khác nhau**   * Giới thiệu Google Cloud Storage * Luyện tập sử dụng các thuật toán khác nhau để tối ưu hiệu quả training. * Đặt giờ tự động pipeline   + Kéo dữ liệu   + Clean dữ liệu   + Tự động training. |
|  |  | **MLOps và những khái niệm liên quan**   * Đọc bài và trả lời các câu hỏi liên quan tới MLOps * Khái niệm MLOps (ML operations) * Tổng quan về End-to-End Machine Learning Workflow * Nguyên tắc MLOps |
|  |  | **CPUs**   * Giới thiệu CPU * Tính toán song song trên CPUs   + Nhiều lệnh song song (ILP)   + Một lệnh trên nhiều luồng dữ liệu (SIMD)   + Đa luồng (Multi-thread parallelism)   + Phân tán (Distributed Compute) * **Đa luồng** tiền xử lý dữ liệu * Bài tập: Lập trình multi threading. |
|  |  | **Chữa bài tập + hướng dẫn thuê/sử dụng VPS CPUs trên Digital Ocean + Google Cloud + AWS**   * Bài tập Pytorch * Bài tập HuggingFace * Bài tập MongoDB * Hướng dẫn thuê/sử dụng CPUs trên Digital Ocean + Google Cloud * Set up pipeline trên VPS   **Deploy CPUs server**   * Kỹ thuật Prefetching trong RAM * Giới thiệu về Flask, Webserver và deploy một model trên CPU |
|  |  | **Cấu trúc Ram**   * L1/L2/L3 Cache * Không gian bộ nhớ vs băng thông bộ nhớ (Memory Locality vs Memory Bandwith) |
|  |  | **CPUs và GPUs**   * Sự khác biệt giữa **CPUs và GPUs** (tối ưu cho độ trễ và tối ưu cho băng thông) * Giới thiệu CUDA * Graphic Pipelines * Demo training trên A100 * Song song trên GPU   **So sánh GPUs và TPUs**   * Điểm mạnh của TPUs so với GPUs * TPU paper**:** https://arxiv.org/pdf/1704.04760.pdf * Turing SM: https://developer.nvidia.com/blog/nvidia-turing-architecture-in-depth/   **Giới thiệu thư viện Numba + Cupy**   * Lập trình song song trên GPUs |
|  |  | **Số thực dấu phẩy động - Floating Point Format**   * Giới thiệu IEEE-754 Floating-Point Standard * TFLOPs   Mix-precision:   * <https://www.tensorflow.org/guide/mixed_precision>   Đọc nghiên cứu   * Deep Learning with Limited Numerical Precision: <http://proceedings.mlr.press/v37/gupta15.pdf> * Map-Reduce for Machine Learning on Multicore: <https://proceedings.neurips.cc/paper/2006/file/77ee3bc58ce560b86c2b59363281e914-Paper.pdf>   **Học máy chính xác thấp (**Low-Precision Machine Learning**)**   * Low-Precision Machine Learning * Brain floating point - Bfloat16 * Mixed Precision Training   Lượng tử hóa hậu training:   * Quantization hậu training với float16 * Quantization hậu động * Quantization hậu training với số nguyên   Lượng tử hóa quan tâm đào tạo  Demo training mô hình với FP16 cùng Tensorflow.  <https://www.tensorflow.org/guide/mixed_precision>  Đọc nghiên cứu:  Quantization and Training of Neural Networks for Efficient  Integer-Arithmetic-Only Inference  <https://arxiv.org/pdf/1712.05877.pdf> |
|  |  | **Kỹ thuật tăng cường hiệu năng mô hình**   * Lượng tử hóa mô hình   + Lượng tử hóa hậu training   + Lượng tử hóa có thể training * Tỉa mô hình (Pruning) * Lượng tử hóa có thể training * Nén mô hình (Model compression) * Chắt lọc tri thức (Knowledge distillation) * Phân cụm tham số (Weight clustering)   Đọc nghiên cứu   * DEEP COMPRESSION: COMPRESSING DEEP NEURAL NETWORKS WITH PRUNING, TRAINED QUANTIZATION AND HUFFMAN CODING   <https://arxiv.org/pdf/1510.00149.pdf>   * WHAT IS THE STATE OF NEURAL NETWORK PRUNING?   <https://proceedings.mlsys.org/paper/2020/file/d2ddea18f00665ce8623e36bd4e3c7c5-Paper.pdf> |
|  |  | **ONNX + TensorRT + Triton Server**   * Thư viện ONNX * Thư viện TensorRT * Triton Inference Server |
|  |  | **Đằng sau Dataset**   * Scan Orders (Sắp xếp dữ liệu tối ưu khi train mô hình học máy trên Ram) * Đánh giá hiệu năng của Tensorflow TF Data Pipeline.   **Xây dựng Crawler/Spider Web**   * Chi tiết về HTML/CSS * Xử lý text từ website * Build Text Dataset |
|  |  | **Đào tạo phân tán (Nguyên tắc 3)**   * Các cơ chế Push/Pull/BroadCast/Reduce/All-reduce/Wait * Parameter server model * Giới thiệu + thực hành tf.distribute.Strategy * Thực hành training mô hình trên nhiều GPUs   **Dữ liệu song song (Data Parallelism)**   * Data Parallelism * Distributed Data Parallelism * ZeRO Data Parallelism |
|  |  | **Sử dụng Docker**   * Sử dụng Docker: Docker Image + Docker Container * Giới thiệu Docker Compose   **Machine learning CI/CD**   * Jenkins * Set up CI/CD sử dụng Travis CI * Auto Deployment |
|  |  | **Google Cloud + Docker cần thiết cho học máy**   * Cloud Storage: Lưu trữ data lớn + Model lớn * Load Balancing: Phân luồng Máy chủ * Compute Engine: Máy chủ * Instance Group: Tự động scaling máy chủ   **Triển khai mô hình trên Amazon Web service.** |
|  |  | **Kubernetes**   * Giới thiệu về Kubernetes * Khái niệm Pods * Lập trình file cài đặt YAML * Networking với Kubernetes * Kiến trúc MicroService * Deploy Kubernetes trên Google Cloud/AWS |
|  |  | **Ml obchestation**   * Giới thiệu về Workflow trong học máy * DAGs là gì * Các framework nổi tiếng để thiết kế Workflow   + Airflow   + DVC |