

Học sâu và ứng dụng trong thị giác máy tính

- Thời gian làm bài: **90mins**
 - Tài liệu: **Được mang theo 01 tờ giấy A4**
 - Hình thức kiểm tra: **trắc nghiệm (chọn một đáp án đúng và đầy đủ ý nhất)**
 - Làm bài: **tô vào tờ trắc nghiệm**
 - Cuối buổi thi: **nộp lại tờ trắc nghiệm và đề thi**
-

• Nội dung ôn tập

1. Về dạng bài toán hay công việc (tasks): **phân loại, hồi quy, phát hiện, phân đoạn, sinh ảnh**. Cho mỗi dạng bài toán, cần hiểu rõ:
 - (1) Phương pháp biểu diễn dữ liệu và nhãn
 - Ví dụ, bài toán **phân loại**:
 - Dữ liệu truyền qua mô hình:
 - khi dùng conv2D, có hình dạng: (N, H, W, C), dùng channel-last hay (N, C, H, W), dùng channel-first;
 - khi dùng conv3D, có hình dạng: (N, D, H, W, C), etc
 - cần biết mã hoá nhãn: dạng văn bản, dạng chỉ số, dạng one-hot là gì?
 - Ví dụ, bài toán **phân đoạn**:
 - mặt nạ (mask) là gì?
 - Ví dụ, bài toán **phát hiện**:
 - boundingbox là gì và bằng biểu diễn như thế nào cho một batch
 - (2) Các mô hình phổ biến: **hãy tìm hiểu nguyên tắc làm việc của các mô hình phổ biến**
 - Ví dụ: cho bài toán phân loại: **VGG, ResNet, DenseNet, MobileNet, InceptionV1, Xception**
 - Ví dụ: cho phân đoạn: **Unet, Yolov8** for instance segmentation
 - Ví dụ: cho phát hiện đối tượng: **Yolov8**
 - Ví dụ: cho sinh ảnh: **VAE, GAN, DIFFUSION**
 - (3) Các hàm mất mát (loss) phổ biến
 - Ví dụ: cho hồi quy: **MSE (L2), MAE(L1), Distribution-Focal-Loss**
 - Ví dụ: cho phân loại: **binary-crossentropy, crossentropy** và các biến thể (**weighted/with-logits**)
 - Ví dụ: cho phân đoạn: **crossentropy, dice-loss, IoU-loss**
 - Ví dụ: cho phát hiện: **IoU-loss** và các biến thể (**Complete-IoU, Distance-IoU**), **Distribution-Focal-Loss**
 - Ví dụ: cho sinh ảnh: **KL-Divergence**
 - (4) Nguyên tắc làm việc của giải thuật huấn luyện

- SGD: **forward, backward, và update** là gì?
- **Regularization** là gì? **weight-decay** là gì?
- (5) Cách ra quyết định cho từng bài toán
 - Ví dụ: với hồi quy: tính ra đáp bằng tính kỳ vọng từ phân phối (đầu ra của mô hình)
 - Ví dụ: với phân loại: **softmax** và sau đó là **argmax** hay **top-k** lớp có confidence cao nhất
 - Ví dụ: với phân đoạn: **softmax** và sau đó là **argmax**
 - Ví dụ: với phát hiện: **non-maxima-suppression**
 - Ví dụ: với sinh ảnh: **sample latent variable + [decoder/gan/reverse]**
- (6) Độ đo (metrics) và cách đánh giá hiệu quả của bài toán
 - Ví dụ: với hồi quy **MSE, RMSE, MAE**, etc
 - Ví dụ: với phân loại: **confusion matrix, precision/recall/f1/accuracy** và **weighted/macro-average**
 - Ví dụ: với phân đoạn: **Dice-Score, IoU-Score**
 - Ví dụ: với phát hiện: **mAP**

2. Về các phép tính toán (lớp tính toán) để xây dựng mô hình học sâu.

- (1) Lớp tuyến tính: Kết nối đầy đủ (**FC**), tích chập (**Convolution**) và các biến thể của Convolution như **Separable** (space vs channel), **DepthWise**, **PointWise**, **Transposed-Convolution**.
 - Nguyên lý làm việc
 - Số tham số có thể học
 - Độ rộng của vùng nhận thức (receptive fields)
 - Hình dạng kích thước của bản đồ đặc trưng đầu vào và đầu ra
 - Công thức liên hệ giữa đầu vào và đầu ra
- (2) Lớp pooling: **max-pooling, average-pooling, global-average-pooling**
 - Nguyên lý làm việc
 - Độ rộng của vùng nhận thức (receptive fields)
 - Hình dạng kích thước của bản đồ đặc trưng đầu vào và đầu ra
- (3) Lớp chuẩn hóa: **batch-norm, layer-norm, instance-norm**
 - Nguyên lý làm việc
 - Số tham số có thể học
 - Độ rộng của vùng nhận thức (receptive fields)
 - Hình dạng kích thước của bản đồ đặc trưng đầu vào và đầu ra
- (4) Lớp phi tuyến (**activation**): **relu, elu, sigmoid, tanh, softmax**
 - Nguyên lý làm việc
 - Số tham số có thể học
 - Độ rộng của vùng nhận thức (receptive fields)

- Hình dạng kích thước của bản đồ đặc trưng đầu vào và đầu ra
- (5) Lớp **Dropout**
 - Nguyên lý làm việc
 - Độ rộng của vùng nhận thức (receptive fields)
 - Hình dạng kích thước của bản đồ đặc trưng đầu vào và đầu ra
- (6) Lớp tổng hợp dữ liệu: **Concat, ElementWise-Add**
 - Nguyên lý làm việc
 - Ràng buộc về hình dạng và kích thước của các bản đồ đặc trưng ở đầu vào và đầu ra
- (7) Lớp/Phép biến đổi hình dạng dữ liệu, hoán vị các chiều trong bản đồ đặc trưng và chia tách bản đồ đặc trưng: **Reshape/View/Flatten; Transpose/Permute; Split/Chunk**
 - Nguyên lý làm việc
 - Ràng buộc về hình dạng và kích thước của các bản đồ đặc trưng ở đầu vào và đầu ra

3. Về transformer và ViT

- (1) Cách chia ảnh thành ô lưới $n \times n$ pixels?
- (2) Positional Embedding là gì?
- (3) Attention là gì và bằng cách nào?
- (4) Nguyên tắc tính toán của từng khối trong transformer
- (5) Số tham số cần học?

4. Về xử lý tín hiệu chuỗi thời gian: **LSTM**

- (1) Nguyên tắc làm việc của một cell LSTM.
- (2) Cách nhiệm các cổng bên trong LSTM, như: forget, input, etc.
- (3) Số lượng tham số cần học.

—hết—