Lab 7: Self-attention and ViT

11210IPT 553000

Deep Learning in Biomedical Optical Imaging

2023/11/06

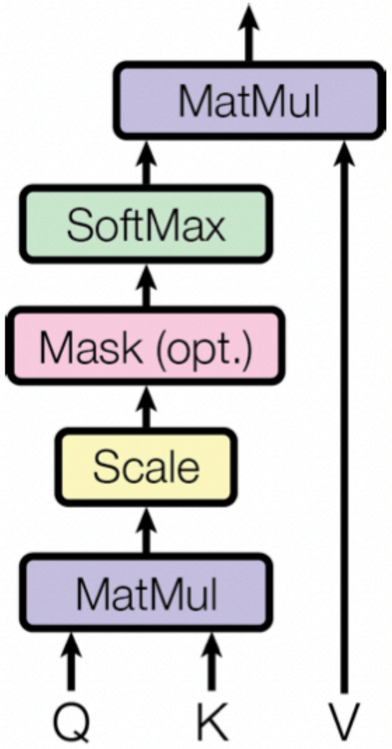
Outlines

- Self-attention
- Vision Transformer (ViT)
- Report

Self-attention

- There are many amazing breakthroughs with self-attention based model in NLP.
- The original idea is from “Transformer” [2]. It proposed a self-attention based model which contains some crucial mechanisms.
  - Scaled dot-product attention
  - Multi-head attention
  - Positional encoding/embedding

Scaled Dot-Product Attention



Query Value

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

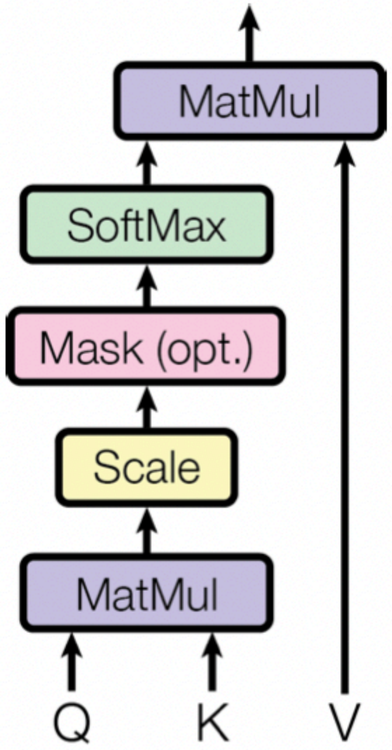
Key

dimension of queries and keys

Why scaled?

When dimension of queries become lager, the dot products grow in magnitude, pushing the softmax function into regions where it has extremely small gradients. So use a salad term to counteract this effect.

Scaled Dot-Product Attention



Query Value

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

Key

dimension of queries and keys

Softmax?

The range of softmax is [0, 1]. The formula is $s\left(x_i\right) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}}$

Scaled Dot-Product Attention

- Why self-attention?
  - The total computational complexity per layer.
  - The amount of computation that can be parallelized, as measured by the minimum number of sequential operations required.
  - The path length between long-range dependencies in the network.

Vision Transformer (ViT)

- Despite the huge successful in NLP, self-attention based models are still hard to beat the convolution-based model in image classification, which is an important task in computer vision…

- …Until the **Vision Transformer** appears! 
  - The architecture of vanilla Transformer [2] and its efficient implementations can be used almost out of the box.
  - The fewest possible modifications to the Transformer design to make it operate directly on images instead of words.

Vision Transformer (ViT)

Classification head

Feature extraction

Position embeddings

Patch embeddings

Image to patches

8

Vision Transformer (ViT)

Classification head

Feature extraction

Position embeddings

Patch embeddings

Image to patches

9

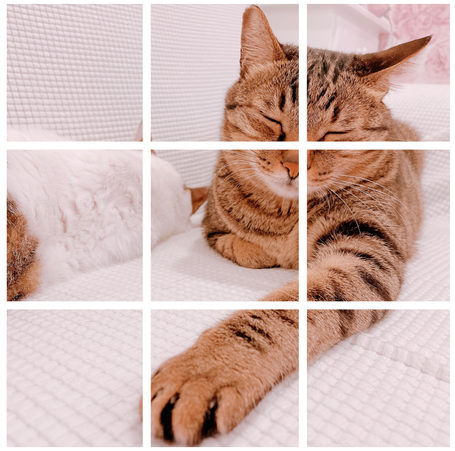
Patches

- ViT divides an image into a grid of square patches.
  - Each patch is flattened into a single vector by concatenating the channels of all pixels in a patch and then linearly projecting it to the desired input dimension.

C: channels

image $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$

HxW: original image



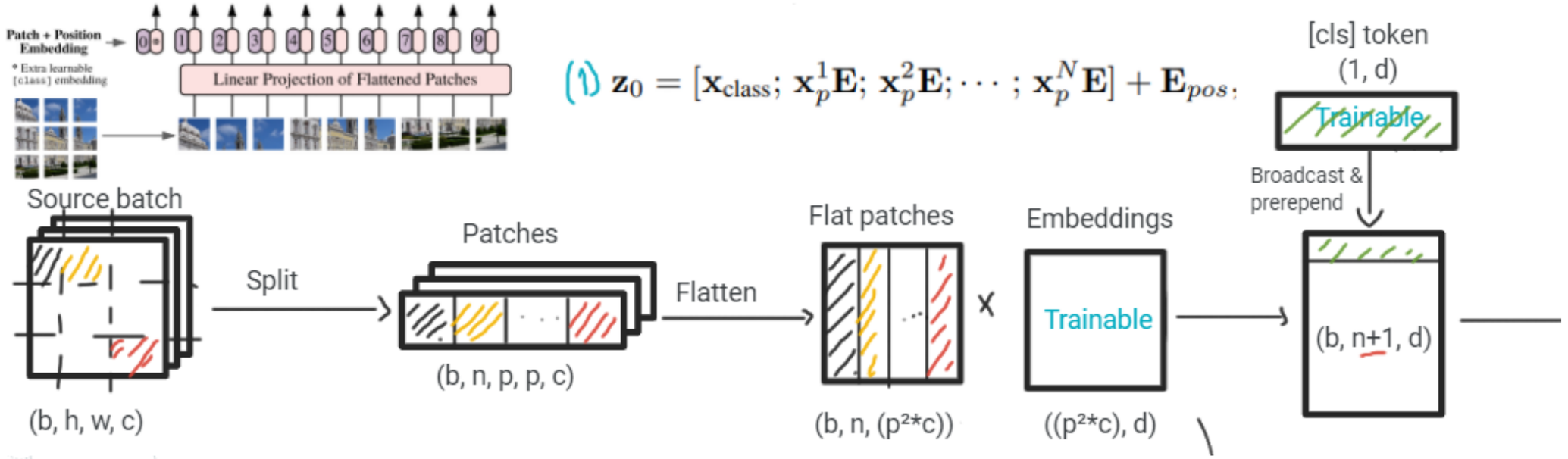
P: image patch

sequence of flattened 2D patches $\mathbf{x}_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$

N: number of patches

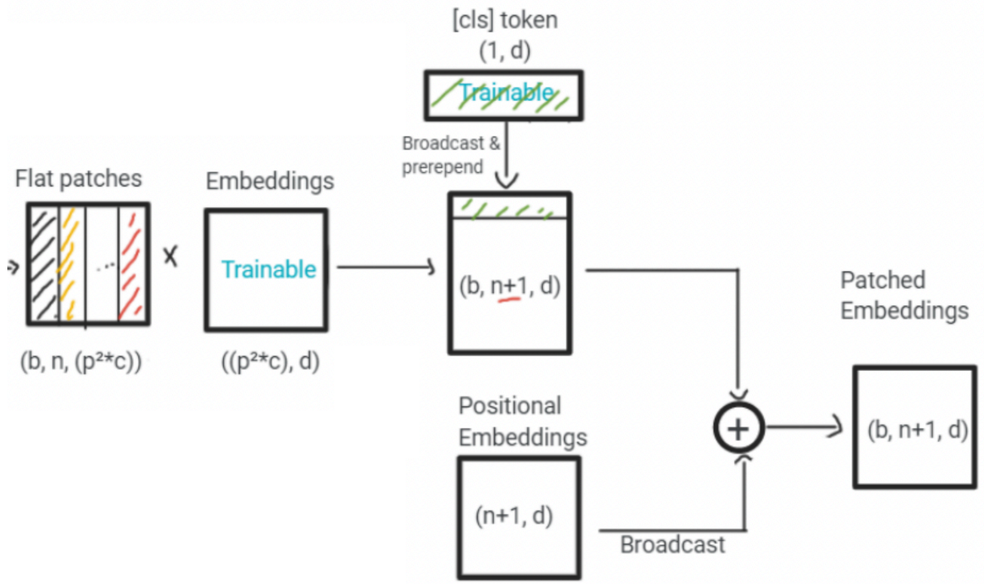
Patch Embeddings

- The Transformer uses constant latent vector size d through all of its layers, so we flatten the patches and map to d dimensions with a trainable linear projection.



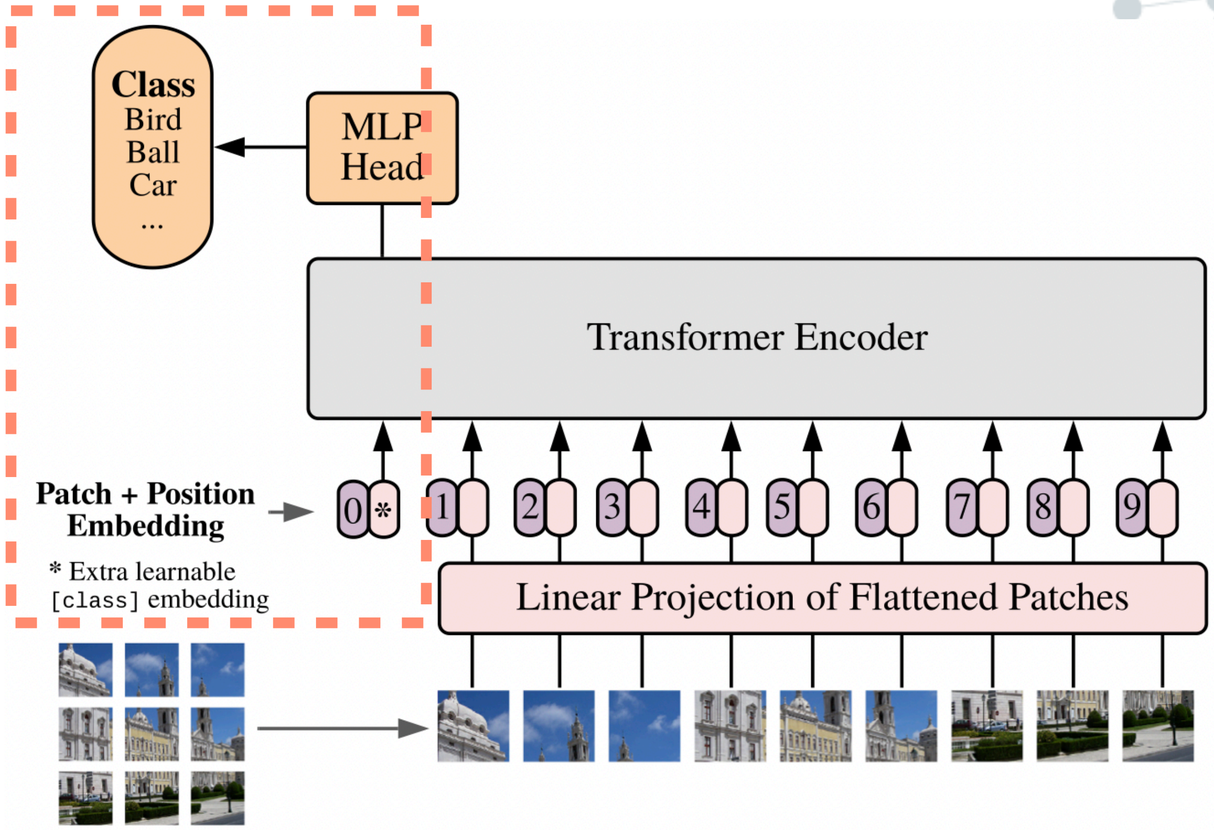
Position Embeddings

- Add learnable position embeddings to each patch, which allow the model to learn about the structure of the images.
  - Help ViT to learn relevant information from the training data and encode structural information.

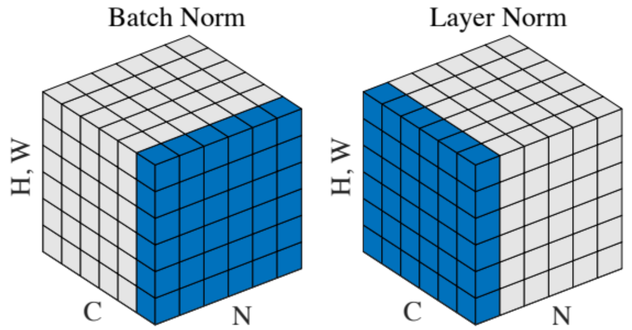


Class Token

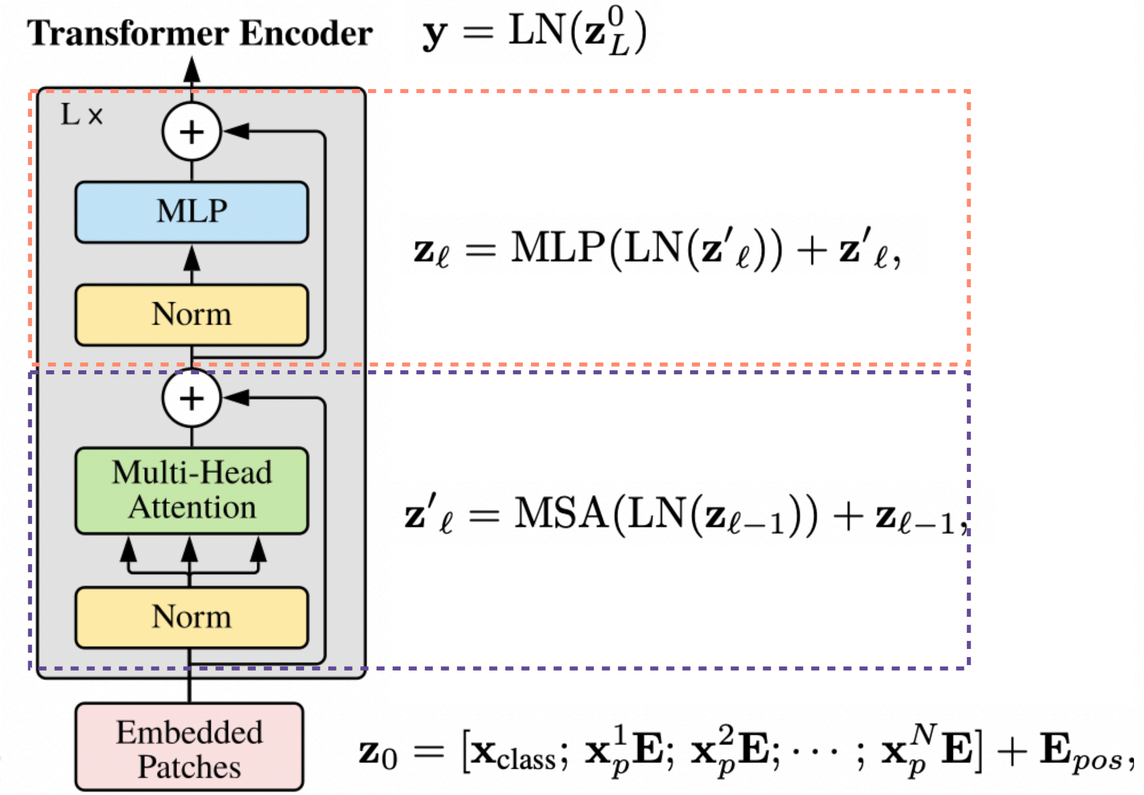
- Inspired from BERT's [class] token [3], add a learnable embedding to the sequence of embedded patches.
- Intuitively, this represents an aggregate of the representations of the patches.
  - Only the last representation corresponding to this token (output of transformer) is fed through the classification layers.

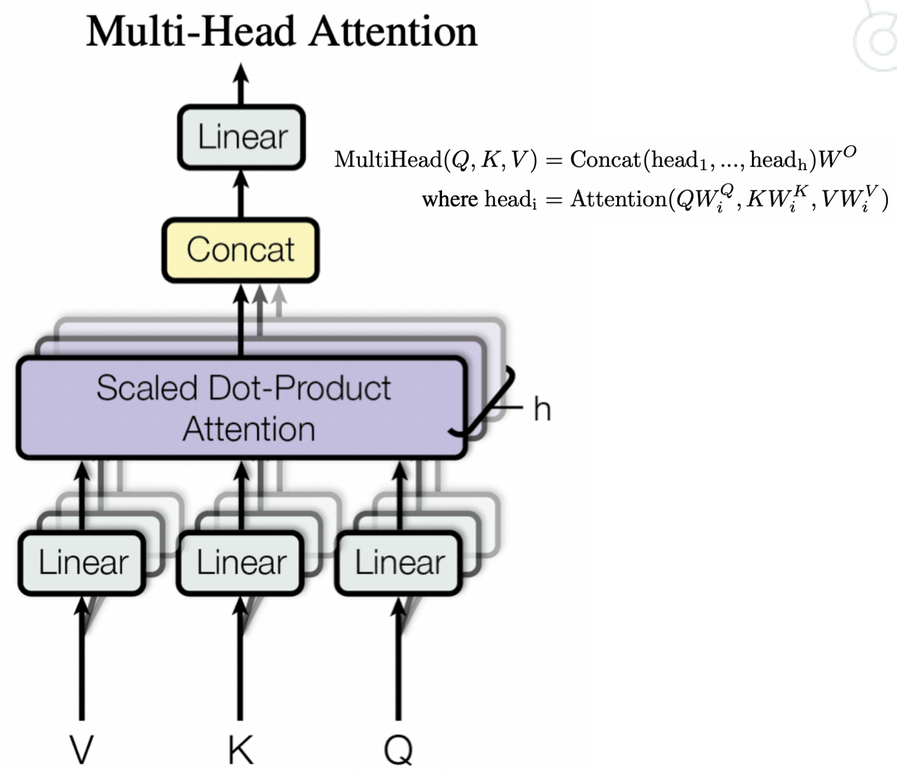


Transformer Encoder

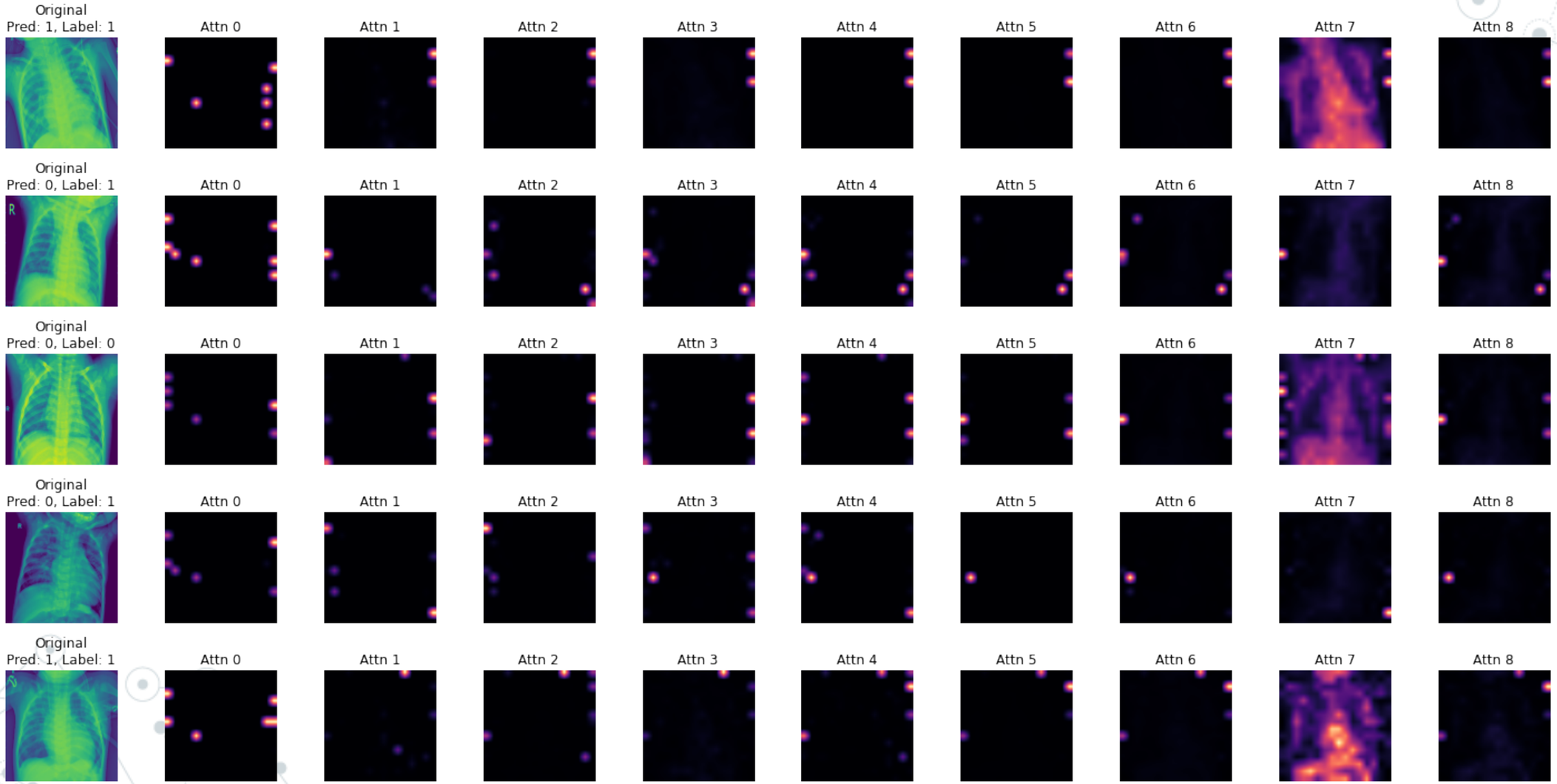


L: # of layers





Attention Map



Report

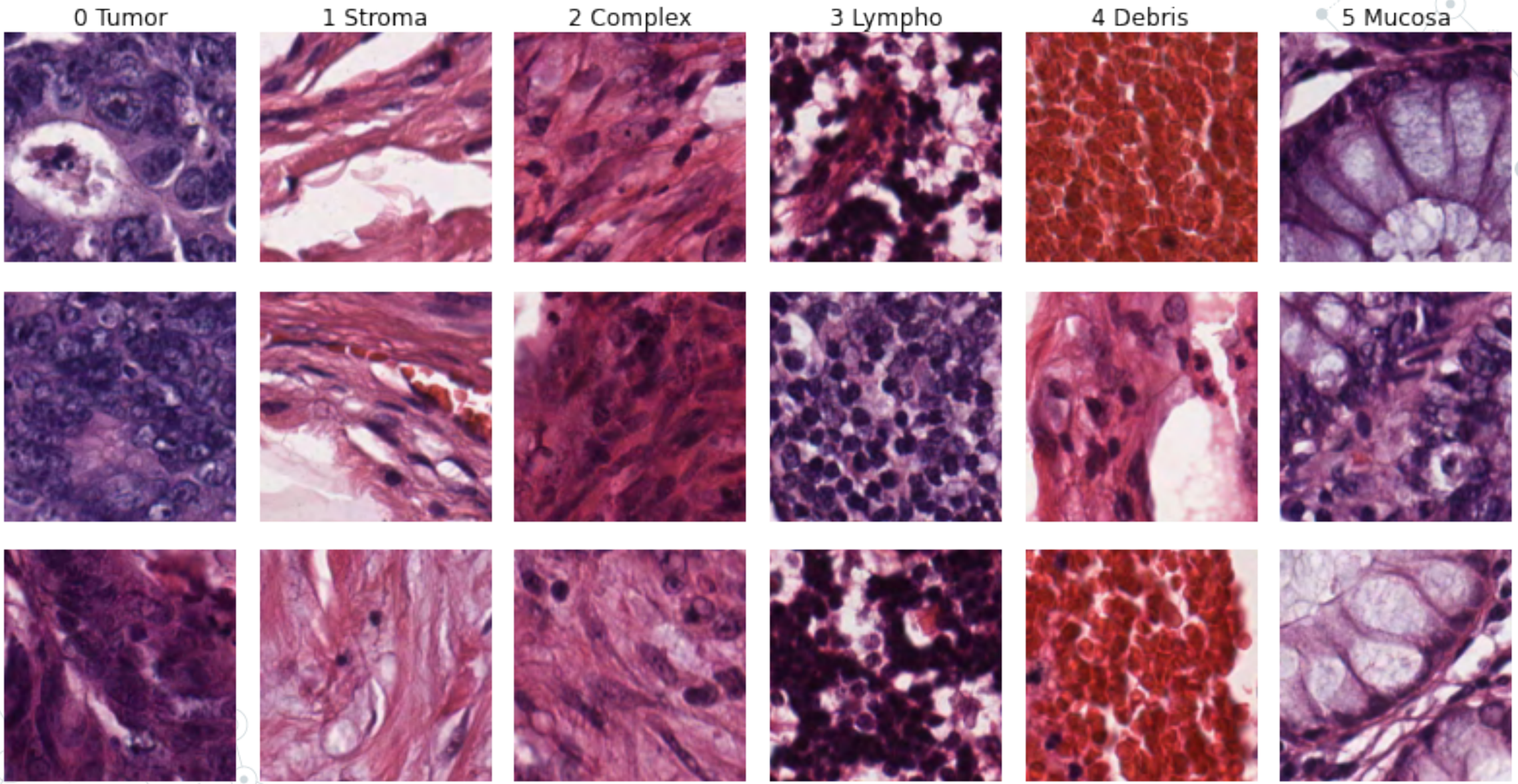
- **Deadline**: 23:59, 4th Dec. (GMT+8)
- We need to write a report to answer questions. Details are in report_description.pdf
- **Important**: Make sure your commit is timestamped before the deadline. Late submissions might not be graded or could incur a penalty. Only the GitHub link is required on NTHU EEclass.

Report - Dataset

- Each sample in the dataset is a 150x150 pixel RGB image representing one of 6 distinct tissue textures that are commonly identified in cancer histology.
  - # of training data: 425 images for each. (Total: 2550)
  - # of validation data: 100 images for each. (Total: 600)
  - # of test data: 100 images for each. (Total: 600)

Report - Dataset

- Hematoxylin and Eosin (H&E) staining is a common method used to examine tissue samples under a microscope. This staining is particularly useful for observing the cellular and tissue structures and identifying pathological changes, such as those related to cancer histology.
  - Hematoxylin stains nucleic acids, making cell nuclei appear blue or purple.
  - Eosin stains the cytoplasm and other structures like collagen fibers, causing them to appear pink or red.


0 Tumor
1 Stroma
2 Complex
3 Lympho
4 Debris
5 Mucosa

Report - Dataset

- Class 0 - Tumor 腫瘤: Refers to an abnormal growth of cells, which can be benign or malignant.
- Class 1 - Stroma 基質: The supportive framework of a tissue or organ.
- Class 2 - Complex 綜合體: A complex arrangement of cells and tissue structures.
- Class 3 - Lympho 淋巴球: Contain lymphocytes that play a crucial role in immune responses.
- Class 4 - Debris 碎片: Typically refers to cell remnants that result from tissue necrosis or damage.
- Class 5 - Mucosa 黏膜: The lining of various bodily cavities and tubes, such as the gastrointestinal tract, which is lined with epithelial cells.

CODING TIME!

References

[1] Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." arXiv preprint arXiv:2010.11929 (2020).

[2] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).

[3] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[4] Brown, Tom, et al. "Language models are few-shot learners." Advances in neural information processing systems 33 (2020): 1877-1901.

[5] Kolesnikov, Alexander, et al. "Big transfer (bit): General visual representation learning." European conference on computer vision. Springer, Cham, 2020.

[6] VTAB Protocol: https://ai.googleblog.com/2019/11/the-visual-task-adaptation-benchmark.html

[7] ViT - open review: https://openreview.net/forum?id=YicbFdNTTy

[8] Understanding the Vision Transformer and Counting Its Parameters

[9] ViT in Paper With Code