DLCV HW3

b07901169 楊宗桓

1 ViT Image Classification

1.1 Accuracy

1.1.1 Different Setting

我實做了not pretrained/pretrained, 不同lr, 不同heads數量, data augmentation(ColorJitter跟RandomHorizontalFlip)

lr	Train Acc	Val Acc
1e-3	93.453	67.764
1e-4	99.688	91.253
5e-5	99.813	92.401
2e-5	100.0	93.839

Table 1: Different lr

num_heads	Train Acc	Val Acc
8	100.0	93.485
12	100.0	93.839
16	100.0	93.248

Table 2: Different num_heads

data augmentation	Train Acc	Val Acc
no augmentation	100.0	93.839
ColorJitter	100.0	92.922
RandomHorizontalFlip	100.0	93.440

Table 3: Different augmentation

由Table.1可以看出lr對model的表現影響很大，當lr過大會使model無法順利學習，無論在training set 或validation set表現都比較差甚至會嚴重overfit。Table.2則可看出heads的數量對Model影響不大，其中以num_heads=12表現最好。而Table.3值得注意的是不管哪一種data augmentation對model都沒有幫助，有可能是因為attention在學習中就可以學到了data augmentation做得到的事，又或者這兩種augmentation其實不接近validation set的image分布。

由Table.4及figure.1可知道pretrain不只可以收斂較快且最後得到的performance也比較好，沒有pretrain的model在train set收斂後validation的acc仍然很低。

pretrained	Train Acc	Val Acc
with	100.0	93.839
without	97.250	12.301

Table 4: Pretrained

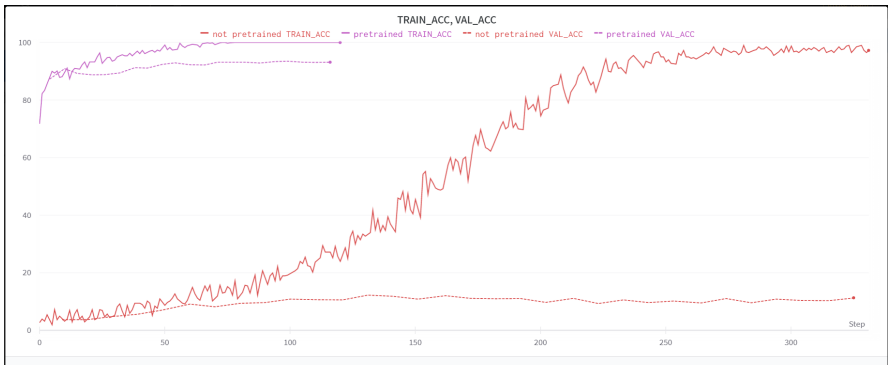


Figure 1: training curve pretrained vs. not pretrained

1.1.2 Best model

我最後使用的model是ensemble + label smoothing, Val acc為***94.0667%***

1.2 Position embedding

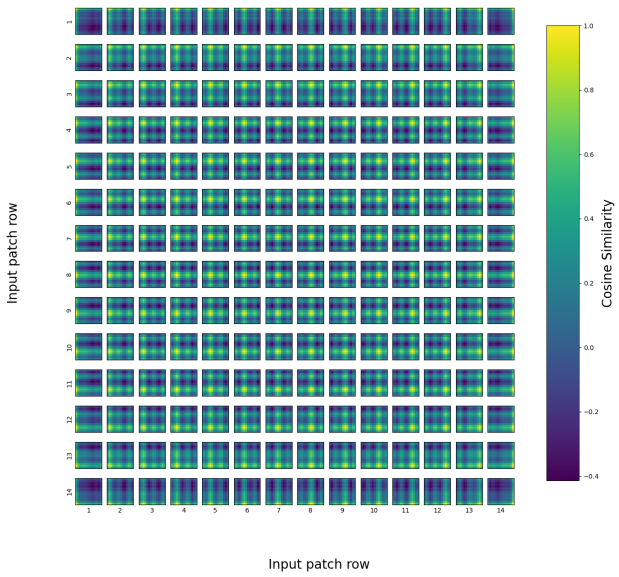


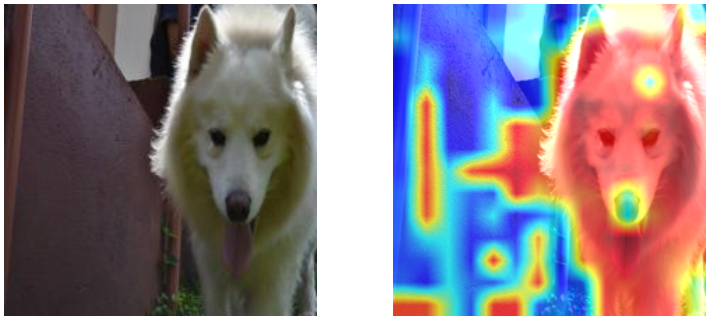
Figure 2: Positional embedding visualization

由figure.2可以看到整張圖大致呈對稱，且每一個position對於同一行或同一列都有較高的Cosine similarity，離該position越近的點也通常有較高的Cosine similarity，表示model有成功學習到position的訊息。

1.3 Attention map

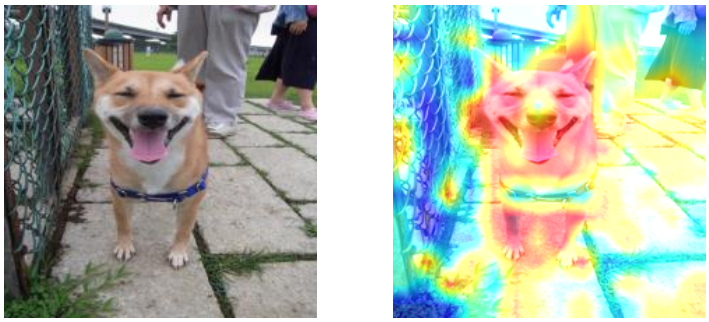
(a) (b)

Figure 3: 26_5064.jpg attention map



(a) (b)

Figure 4: 29_4718.jpg attention map



(a) (b)

Figure 5: 31_4838.jpg attention map

由figure.3 - figure.5可以看到雖然有部分地方attend到背景(如figure.3貓周圍的黃/紅點)但大部分背景的attention值都較低(藍色)，而attention值高的地方還是有大部分對到pet所在的位置，代表model有運用attention學到如何辨識pet的特徵及位置進而做分類。

2 ViT Image Captioning

2.1 Attention map

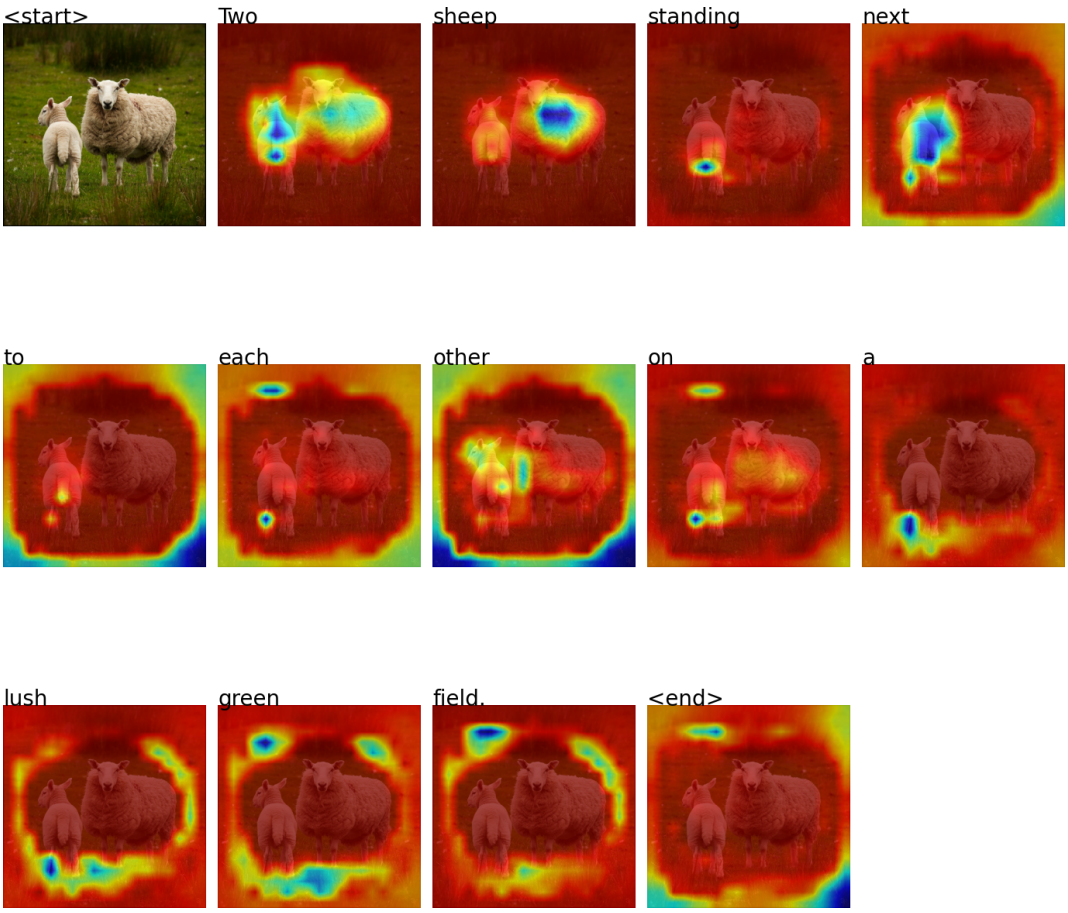


Figure 6: sheep.jpg attention map

產生的caption與圖片內容一致且文法與用字也沒有錯誤。attention到的地方也有對到物體所在的位置，例如”two”attend到兩隻綿羊的位置，而”sheep”attend到sheep所在的位置，”lush”、”green”、”field”也都有attend到綿羊周遭也就是草地。

2.2 What I learn

最難的地方在於怎麼從transformer取出attention weight花了很多時間track transformer的output過程結果發現nn.MultiHeadAttention return的output就有attention weight。另外我也發現attention map中介係詞attend到的地方其實跟他前後的名詞attend到的地方，有可能是因為他們是以一起代表圖片某部分的訊息所以attention的位置也差不多。