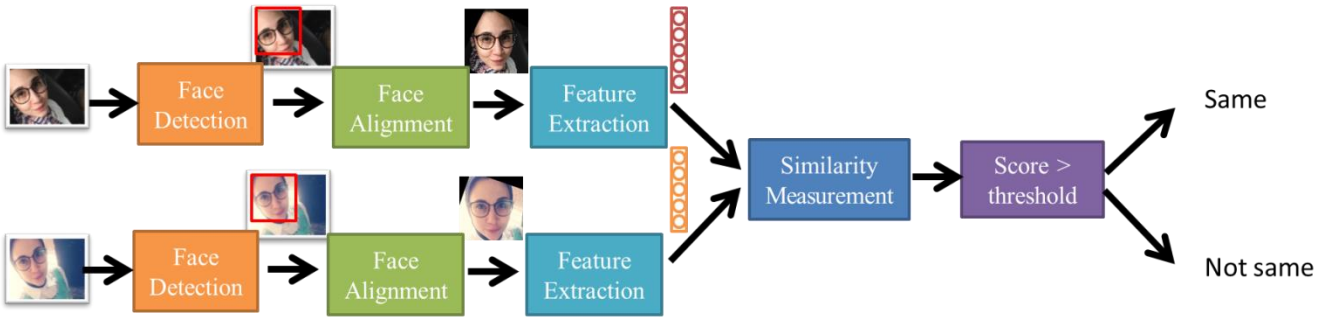
AMMAI HW1: APD Face Verification

R08944004 萬世澤

在這次作業中，主要實驗基於三種不同的 cost function 在亞洲政治人物資料集的人臉辨識能力，實驗以 Face Verification 的形式進行，其中實驗指標將使用 ROC、AUC。另外，實驗將比較有無使用 normalization layer、distance function 選用、跨人種測試集對模型辨識能力之影響等議題，測試架構如下：



▲ Face Verification 架構概要

執行程式

作業主要以 Python 與 Pytorch 完成，安裝說明、程式執行與實驗結果等相關資訊，請見 readme.md。

實驗設定

訓練資料

訓練影像主要使用 CASIA-WebFace 資料集[1]，其中我是選用其經過人工資料清洗後的版本，其具有 455594 張人臉影像，共 10575 個人臉類別。選用其作為訓練集主要有以下考量：

1. 該資料集已進行人工清理
2. 設備資源有限：有別於其他更大的資料集如 VGG-FACE2(約 300 萬張、8631 類別)，在算力有限的情況下，將便於測試更多方法。再者，據文獻[2]指出，利用具有較多類別的資料集訓練淺層模型將有助於模型性能提升。

影像前處理

輸入臉部影像會先透過人臉偵測(face detection)與人臉校正(face alignment)進行前處理，人臉偵測部分會透過 MTCNN[3]進行偵測與裁切，然後根據 MTCNN 所產生的五個臉部標記點進行人臉校正，其方法為利用預先定義的五個臉部標記點位置當作對應點，分別為左右眼、鼻和左右嘴角，再把臉部影像的五個標記點與對應點輸入進行仿射轉換(Affine transformation)，藉以計算此臉部影像要移動、旋轉、放大或縮小若干尺度才能變正臉部影像。該階段實作主要使用[4]。

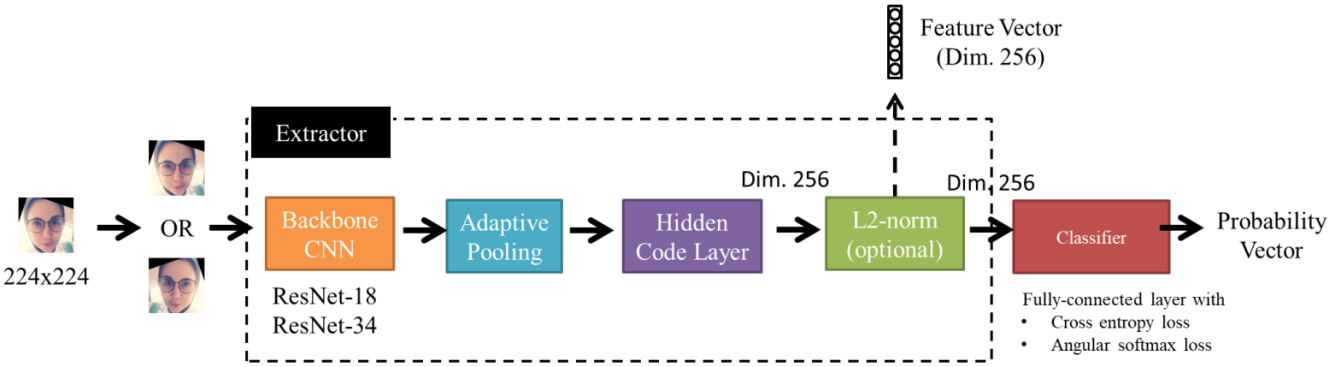
此外，由於前處理階段需將資料以 MTCNN 進行人臉偵測，若模型判斷畫面上無人臉，實驗將直接將該影像丟棄，以下是處理後之結果：

	Before MTCNN	After MTCNN
Clean CASIA WebFace (Train Set)	455594 images	452270 images
APD (Test Set)	21796 pairs	20227 pairs

在訓練階段，校正後的人臉影像除進行正規化[1]外，還會以水平翻轉進行資料擴增，其中選用水平翻轉是因為人臉左右翻轉後應具不變性，再者考慮其便利性故選用該方法。而在測試階段，僅需套用相同的正規化方法即可。

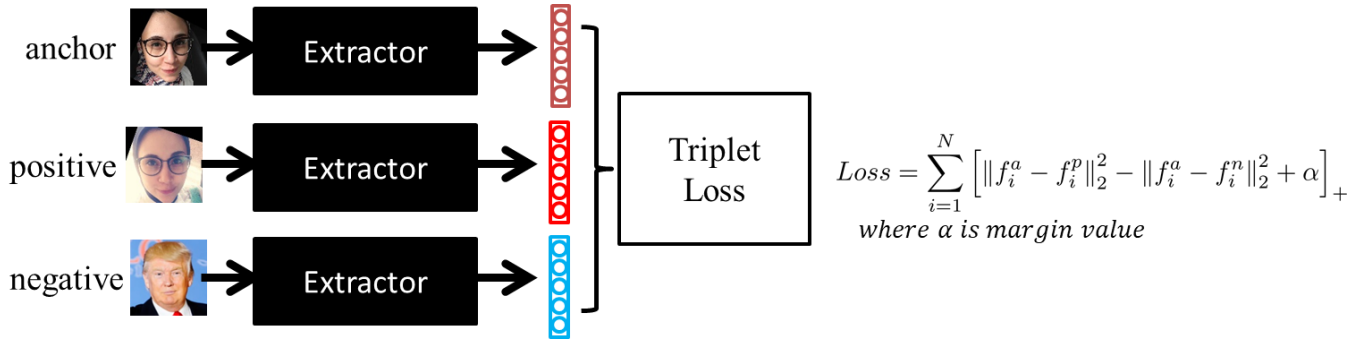
模型架構

模型架構主要基於 ResNet-18 與 ResNet-34 進行，校正後的影像會先經過 CNN 抽取特徵並配合自適應池化(adaptive pooling)降低到固定的維度，之後經過 Linear Transformation(即下圖的 Hidden Code Layer)來獲得特徵向量。在基於 Softmax-only、Angular softmax[5]的架構下，特徵向量主要作為學習分類任務的附帶功能，特徵向量會後送至分類器產出各標籤預計機率向量(probability vector)，其維度根據資料集的類別數而定，以本次作業為例，其維度為 10575，而特徵向量的維度為 256。



▲ 以學習分類任務取得影像低維特徵

在 Metric learning 架構下，模型直接以 pair 或 triplet[6]的形式學習特徵向量，藉以做為該人臉的低維表示法，以本次作業為例，其維度為 256。



▲ Metric Learning: Triplet Loss 示意圖

[1] 正規化方法與 Pytorch 官方訓練 ImageNet 的方法相同，詳情見：
https://github.com/pytorch/examples/blob/97304e232807082c2e7b54c597615dc0ad8f6173/imagenet/main.py#L197-L198

訓練方式

所有模型將以 SGD 進行訓練，並將初始 learning rate 設為 0.1，其中基於 Softmax-only、Angular-Softmax 學習的模型將以每 20 個 epoch 降低 1/10 的方式進行學習率的調整，此外，所有的實驗設定都將利用 ResNet-18、ResNet-34 各學習 50 個 epoch，詳情如下表。

	Softmax-only、Angular Softmax	Triplet Loss
Batch size	256	128
Learning rate	#Epoch < 20 => 0.1 20 < #Epoch < 40 => 0.01 40 < #Epoch < 50 => 0.001	0.1 (constant learning rate)
Optimizer	SGD	SGD
Remark	在訓練階段，Softmax-only 不對 input 進行 l2-norm，而 Angular Sofmax 會對 weight 與 input 進行 l2-norm Angular Softmax 設為 cos(3θ)	在訓練階段，Triplet Loss 固定使用 L2-norm Margin term alpha 設為 0.5

其中，基於 Triplet Loss 的設定與前二者在 bath size 與 learning rate 稍有不同，triplet loss 在 forward 過程中需要建立三次 backpropagation path，使用過大的 batch size 將導致 GPU memory 不足。另外，learning rate 將不進行調整，根據實驗 learning rate 恆定為 0.1 能獲得更好的表現，我想這部分跟 hard sample mining 的學習過程有很大的關係。

在實作的部分，Angular Softmax 使用[4]提供的實作，Triplet Loss、ROC Curve 使用[7][8]的實作。

實驗結果

實驗將在三個測試集進行，分別是

1. 作業提供裁減過後的 APD 資料集，圖片大小 256x256，共 21796 pairs
2. 使用前述前處理方法的 APD 資料集，圖片大小 224x224，共 20227 pairs
3. 使用前述前處理方法的 LFW 資料集，圖片大小 224x224，共 6000 pairs

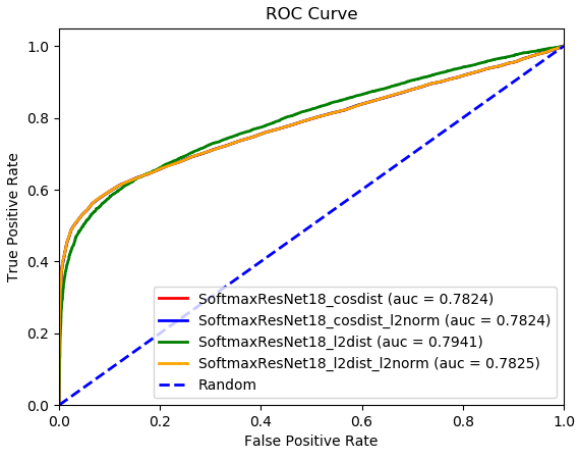
主要針對以下議題進行實驗：

1. 是否採用 L2-norm layer 於**測試**階段
2. 特徵距離計算使用 L2 distance 或是 Cosine distance
3. 跨人種資料集是否影響模型預測能力
4. **訓練**階段加入 l2-norm 對模型預測能力之影響

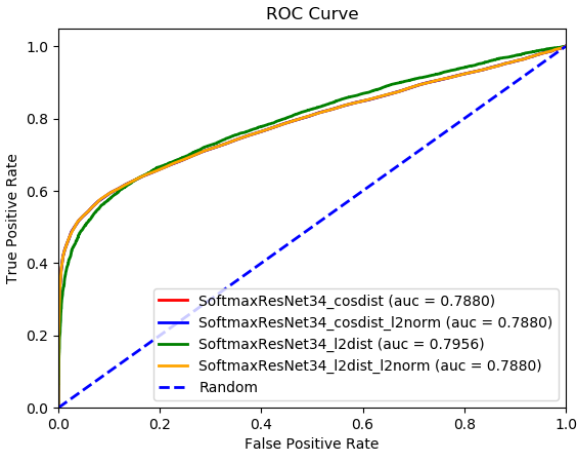
測試用的 checkpoint 將會選擇在 APD 224x224 上表現最好的那個 epoch。

實驗一：採用 L2 normalization layer 與不同 distance function 於測試階段之比較

Softmax-only (測試在 APD 224x224)

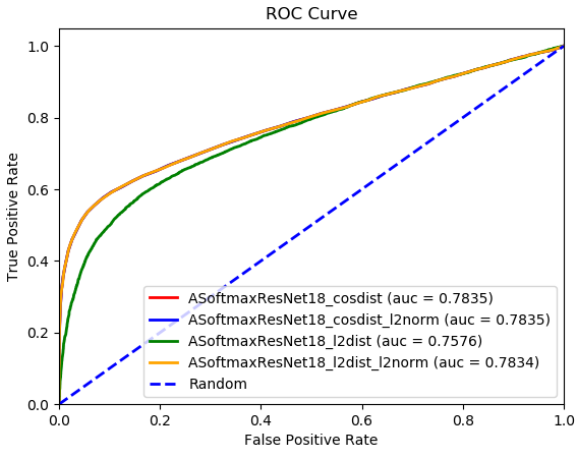


ResNet-18 with Softmax

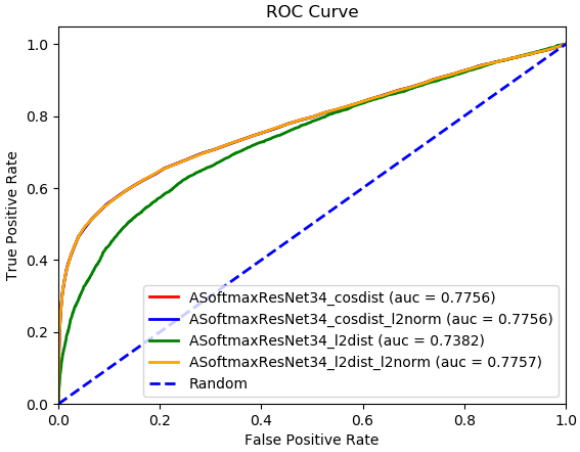


ResNet-34 with Softmax

Angular Softmax, ASoftmax (測試在 APD 224x224)

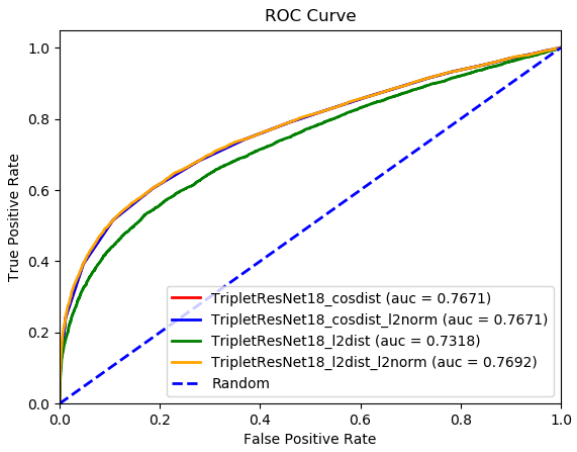


ResNet-18 with Angular Softmax

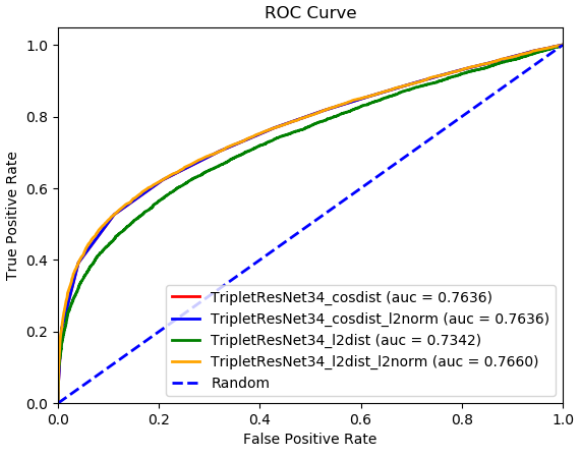


ResNet-34 with Angular Softmax

Triplet Loss (測試在 APD 224x224)



ResNet-18 with Triplet Loss



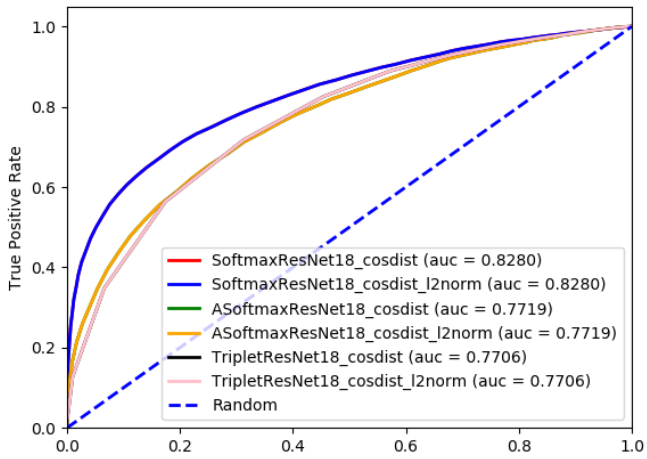
ResNet-34 with Triplet Loss

圖示說明：[architecture]_[distance]_[norm]，[architecture]代表模型架構、[distance]代表距離函數、[norm]代表是否使用 l2 normalization，[distance]可為 l2dist(l2 distance)與 cosdist. (cosine distance)。

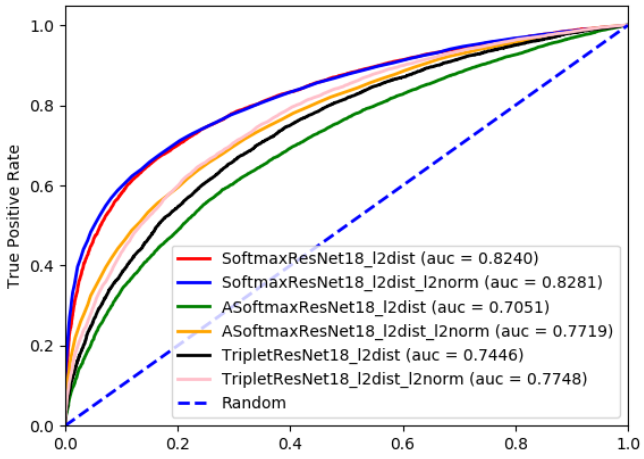
討論：在 normalization 方面，Cosine distance 不受 l2 normalization layer 影響，而 L2 distance 受 l2 normalization 影響較為明顯。在 distance 方面，Softmax-only 搭配 l2 distance 表現較佳，反之，基於 Angular Softmax、Triplet Loss 上額外加入對 input 的 l2 normalization 將會損害 AUC (Area Under Curve)。

實驗二、跨人種資料集預測能力比較

APD (256x256)

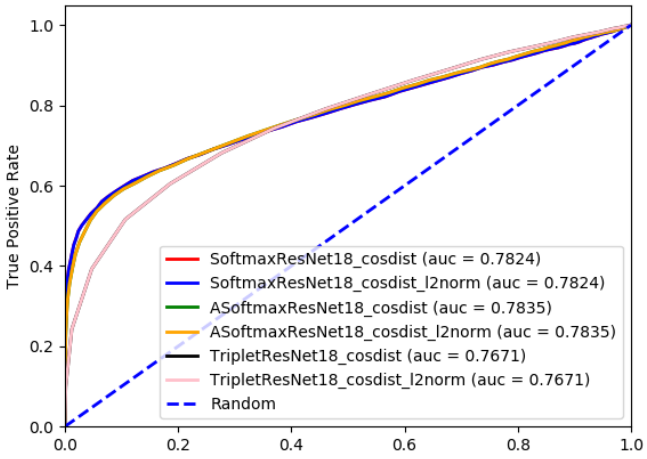


ResNet-18 with cosine distance

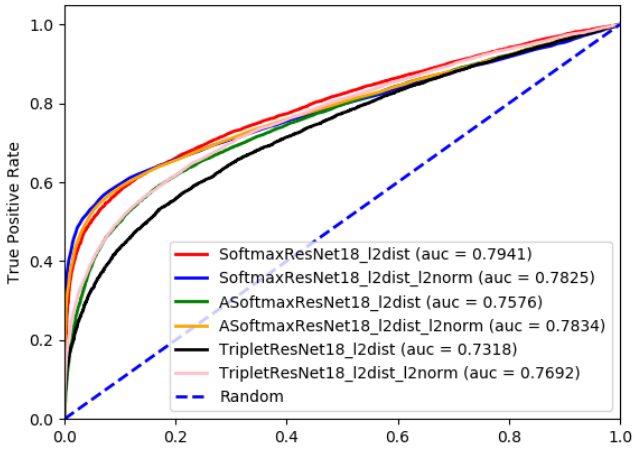


ResNet-18 with L2 distance

APD(224x224)

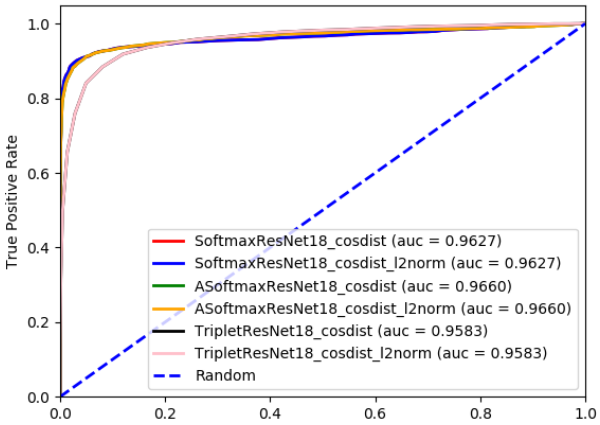


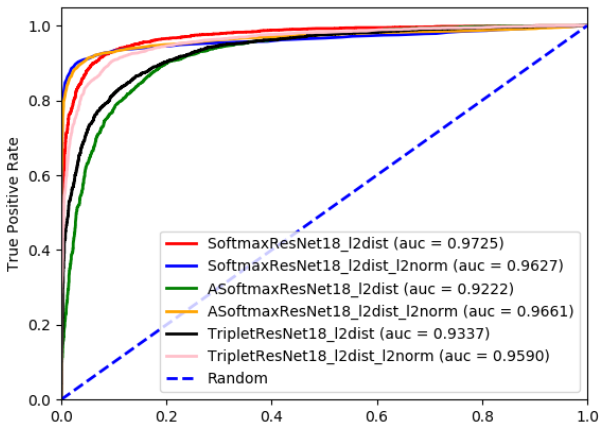
ResNet-18 with cosine distance



ResNet-18 with L2 distance

LFW(224x224)





討論：在最佳結果方面，APD 256x256 的最佳 AUC 為 0.8281 (SoftmaxResNet18_l2dist_l2norm)、APD 224x224 的最佳 AUC 為 0.7941(SoftmaxResNet18_l2dist_l2norm)、LFW 224x224 的最佳 AUC 為 0.9725(SoftmaxResNet18_l2dist)，綜觀而言，Softmax 搭配 L2 normalization 似乎就能達到最佳結果。另外，在 LFW 與 APD 資料集之間可以發現有著巨大的差異，由於大部分的人臉訓練集都集中在白人人臉，對於亞洲人臉的紀錄甚少，可以看出 cross domain 仍然是一個難題(LFW 0.97 vs APD 0.83)。再者，在 APD 224x224 與 APD 256x256 方面，train 在 224x224 測試在 256x256 似乎並沒有一個明顯的優勢，如 Softmax-only 會變得更好 (0.7941->0.8281)，而 ASoftmax 會變差(0.7835->0.7719)，我的想法是 ASoftmax 的 margin 似乎 overfit 在固定的尺寸上，所以當尺度放大時，原本的 margin 將不適用。
最後，在 3 種 cost function 上的比較，我發現 ASoftmax、Triplet Loss 並沒有明顯的優勢，在 ASoftmax 多了一個參數 m 需要調整，若 m 開得過大，則 loss 會變得 NaN，可能是因為 mθ 超過單位圓(>360)。在 Triplet Loss 方面，可能是因為沒辦法挑到夠多具鑑別力的 triplet 所以 AUC 表現不佳。

實驗三、訓練階段是否使用 l2 normalization layer (測試階段不使用 l2-norm)

	Softmax	Softmax w/ l2 norm	ASoftmax	ASoftmax w/ l2norm
ResNet-18	0.7824	0.6464	0.7835	0.6971
ResNet-34	0.7880	0.6423	0.7756	0.6933

討論：在訓練階段配 l2 normalization 在 feature 上，在大部分的情形皆表現不佳，且我發現加入該項將使得收斂速度變得緩慢，所以未來實驗應盡量避免使用。

參考資料

[1] Yi, Dong, et al. "Learning face representation from scratch." arXiv 2014.
[2] Ranjan et al., “Deep Learning for Understanding Faces: Machines May Be Just as Good, or Better, than Humans”. IEEE Signal Processing Magazine 2018.
[3] Zhang, Kaipeng, et al. "Joint face detection and alignment using multitask cascaded convolutional networks." IEEE Signal Processing Letters 23.10 (2016): 1499-1503.
[4] https://github.com/ZhaoJ9014/face.evoLVe.PyTorch
[5] Liu, Weiyang, et al. "Sphereface: Deep hypersphere embedding for face recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.
[6] Schroff, Florian, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.
[7] https://github.com/tamerthamoqa/facenet-pytorch-vggface2
[8] https://github.com/liorshk/facenet_pytorch