Machine Learning

Project3 report

顾涵雪 3160101780

顾钰峰 3160104444

钱泽铖

2018 年 11 月 22 日

目录

1 整体思路 3

2 网络介绍 3

2.1 ResNet50 网络 4

2.2 VGG16 网络 5

2.3 孪生网络 (Siamese Network) 7

2.3.1 网络简介及主要思想 7

2.3.2 网络结构 8

2.3.3 损失函数定义 8

2.3.4 优点 9

2.4 三元组损失函数 (Triplet Loss) 10

2.4.1 函数介绍及主要思想 10

2.4.2 在线三元组构造 (Online Triplet Mining) 12

3 实现过程 12

3.1 数据处理 12

3.1.1 Siamese 网络数据处理 12

3.1.2 Resnet 和 VGG 网络数据处理 13

1　整体思路

我们需要解决的是一个**人脸验证**问题，即输入两张图片，判断他们是否来自同一个人。**人脸验证**问题可以转换为**人脸识别**问题来解决。但是我们的任务也有一定的特殊性，即样本的类别数很多，有一万多种不同的人脸类别，每个类别里的样本数量又比较少。如果按照**人脸识别**问题来处理，必须明确的指出每张图片属于哪一类，由于数据集太大，我们的时间和设备有限，训练效果可能不是太好。而**人脸验证**则无需明确指出图片属于哪一类，只需要对两张人脸进行体征提取后比较得出两张图片是否对应同一个人。

2　网络介绍

在这个问题中，我们尝试了两种思路。

- 第一种思路是将该**人脸验证**问题转化为**人脸识别**问题来处理。通过神经网络来训练分类器，区分图片类别。再将测试集放入分类器，不需要得到最终的分类结果，只需要提取出用于分类的特征向量，通过比较从两张图中提取出的特征向量的差异（如欧氏距离）来判断它们是否来自于同一个人。针对这种方法我们分别尝试了 ResNet50 和 VGG16 两种网络结构，使用 Softmax 作为损失函数。这两种网络都是当前比较流行的网络结构，在解决分类问题上也都十分有效。由于 Resnet50 层数比较深，数据量又很大，我们在有限的时间和硬件平台上没有实现网络较好的收敛?；而 VGG16 可能由于网络层数较浅，虽然在训练集上收敛性很好，但可能出现了过拟合的情形?。

- 另一种思路是采用针对**人脸验证**问题所提出的 Siamese 网络，这个网络在斯坦福大学 cs231n 课程中有详细的讲解。它采用的方法是从数据中映射一个相似度空间，是的同一类别的图像之间的距离尽可能地小，不同类别的图像之间的距离尽可能地大。我们借鉴了 minist 手写数字识别的 Siamese 网络模型，将原本的 2D 网络换成了 3D 网络，并采用 VGG 网络进行降维处理提取图像特征，并使用 Triplet Loss 替换 Contrastive Loss 作为损失函数，这种方法在解决该人脸验证问题上有较好的结果?。

2.1　ResNet50 网络

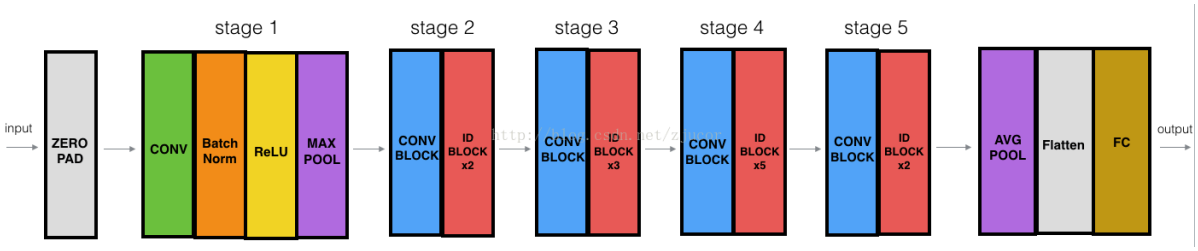

图 1: ResNet50 网络结构

ResNet 网络在设计时考虑到卷积层不同深度所提取特征的尺度差异性，将其融合起来作为提取的特征向量。它有 2 个基本的 block，一个是 Identity Block，输入和输出的维度是一样的，所以可以串联多个；另外一个基本的 block 是 Conv Block，因为输入和输出的维度不一样，所以不能继续串联，它的作用是改变特征向量的维度。

ResNet 网络有两种 mapping 的方式：

- 一种是 identity mapping，就是本身 x。
- 另一种是 residual mapping 是指”残差“，即 $F(x)$, 最后的输出就是 $y = F(x) + x$。

ResNet 网络可以解决随着网络加深，准确率下降的问题。即如果网络已经达到最优，继续加深网络，residual mapping 将被置为 0，只剩下 identify mapping，这样理论上网络就一直处于最优状态了。我们选择的是 ResNet50 的结构，其具体的结构如图所示。

表 1: Resnet50 结构

layer name	50-layer
conv1	7×7,64,stride2
conv2_x	$\begin{pmatrix} 1\times1 & 64 \\ 3\times3 & 64 \\ 1\times1 & 256 \end{pmatrix} \times 3$
conv3_x	$\begin{pmatrix} 1\times1 & 128 \\ 3\times3 & 128 \\ 1\times1 & 512 \end{pmatrix} \times 4$
conv4_x	$\begin{pmatrix} 1\times1 & 256 \\ 3\times3 & 256 \\ 1\times1 & 1024 \end{pmatrix} \times 6$
conv5_x	$\begin{pmatrix} 1\times1 & 512 \\ 3\times3 & 512 \\ 1\times1 & 2048 \end{pmatrix} \times 3$
	average pool
	softmax

在网络的最后再加上一个 average pooling，得到 2048 维特征向量，我们用向量之间的差异性来体现图片之间的相似性（欧式距离和余弦相似度的加权）作为衡量标准来调整参数。实验表明，使用余弦相似度比欧式距离效果更好 [2]，我们为了稳妥起见，使用权 ω 来进行加权。

$$J = ||f(X_1) - f(X_2)||_2 + \omega|cos(f(X_1), f(X_2))|$$

2.2　VGG16 网络

VGG 网络是在 AlexNet 网络的基础上发展而来的，其主要贡献在于使用了非常小的 3*3 的卷积核进行网络设计，并且将网络深度增加到 16-19 层，可以较好的处理分类任务。它把网络分为 5 层，用了 3×3 的过滤器，组合起来作为卷积序列进行处理,

它的结构如图所示。

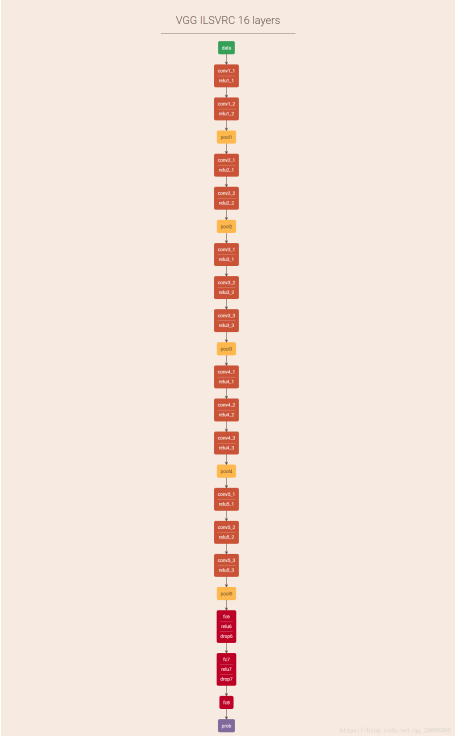

图 2: VGG16 网络结构

VGG16 网络的主要特点包括：

- 网络更深，DCNN、channel 数目更大。
- 采用多个 3*3 的卷积核，可以达到更大的感受野的效果。

其实作为分类模型，它的实际性能已经比不上后来的 ResNet，但是由于在这次任务当中的时间与硬件的限制以及较大的数据量，我们的 ResNet 网络没有在整个训练集上完全训练完成，仅仅收敛到 65% 左右，而 VGG 网络的收敛速度更快，尤其是当我们去除最后两个全连接层后，可以达到更好的收敛效果。考虑到数据类别多，各个类别的训练样本又较少，时间较紧，采取 ResNet 分类器的训练效果可能不是很好，所以搭建较为浅层的 VGG 网络来作为分类器的代表来跟我们主要训练和测试的 Siamese 网络的效果进行对比。

2.3　孪生网络 (Siamese Network)

孪生网络 (Siamese Network) 是一种相似性度量方法，当类别数多，但每个类别的样本数量少的情况下可用于类别的识别、分类等，于 2005 年发表在 CVPR 上的论文 Learning a Similarity Metric Discriminatively, with Application to Face Verification 中首次被提出。

2.3.1　网络简介及主要思想

用于区分人脸的传统分类方法是需要确切的知道每个样本属于哪个类，需要针对每个样本有确切的标签。而且相对来说标签的数量是不会太多的。当类别数量过多，每个类别的样本数量又相对较少的情况下，这些方法就不那么适用了。对于整个数据集来说，数据量足够，但是对于每个类别来说，可以只有几个样本，那么用分类算法去做的话，由于每个类别的样本太少，很难训练出较好的结果，所以只能去找个新的方法来对这种数据集进行训练，从而提出了 Siamese Network。Siamese 在英文中取“孪生”、“连体”的意思。Siamese Network 从数据中去学习一个相似性度量，用这个学习出来的度量去比较和匹配新的未知类别的样本。这个方法能被应用于那些类别数多或者整个训练样本无法用于之前方法训练的分类问题。

Siamese Network 是通过一个函数将输入映射到目标空间，在目标空间使用简单的距离（如欧式距离）进行对比相似度。在训练阶段最小化来自相同类别的一对样本的损失函数值，最大化来自不同类别的一堆样本的损失函数值。给定一组映射函数 $G_\omega(X)$，其中参数为 ω，我们的目标就是去找一组参数 ω，使得当 X_1 和 X_2 属于同一个类别的时候，相似性度量

$$E_\omega(X_1, X_2) = \|G_\omega(X_1) - G_\omega(X_2)\|$$

是一个较小的值，当 X_1 和 X_2 属于不同的类别的时候，相似性度量 $E_\omega(X_1, X_2)$ 较大。这个系统是用训练集中的成对样本进行训练。当 X_1 和 X_2 来自相同类别的时候，最小化损失函数 $E_\omega(X_1, X_2)$，当 X_1 和 X_2 来自不同类别的时候，最大化 $E_\omega(X_1, X_2)$。这里的 $E_\omega(X)$ 除了需要可微外不需要任何的前提假设，因为针对成对样本输入，这里两个相同的函数 $G_\omega(X)$，拥有一份相同的参数 ω，即这个结构是对称的，我们将它叫做孪生结构 (Siamese Architecture)。在这篇论文中，作者用这个网络去做面部识别，比

较两幅图片是不是同一个人，而且这个网络的一个优势是可以去区分那些新的没有经过训练的类别的样本。

2.3.2 网络结构

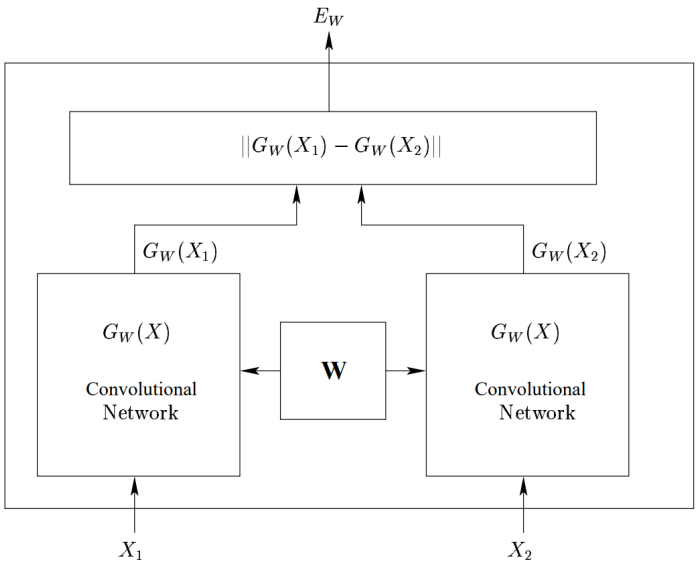

图 3: 孪生网络结构 (Siamese Architecture)

从网络结构图中可以看出，左右两边两个网络具有完全相同的网络结构，它们共享相同的权值 ω，输入数据为一对图片 (X_1, X_2, Y), 其中 $Y = 0$ 表示 X_1 和 X_2 属于同一个人的脸，$Y = 1$ 则表示不为同一个人。即相同对为 $(X_1, X_2, 0)$, 欺骗对为 $(X_1, X_2', 1)$ 针对两个不同的输入 X_1 和 X_2，分别输出低维空间结果为 $G_\omega(X_1)$ 和 $G_\omega(X_2)$，它们是由 X_1 和 X_2 经过网络映射得到的。然后将得到的这两个输出结果使用函数 $E_\omega(X_1, X_2)$ 进行比较。[1]

2.3.3 损失函数定义

为了简化表达，用 E_ω^G 来表示 $E_\omega(X_1, X_2)$，用 E_ω^I 来表示 $E_\omega(X_1, X_2')$。假设损失函数仅与输入输出有关，则可定义为

$$\Gamma(\omega) = \sum_{i=1}^{P} L(\omega, (Y, X_1, X_2)^i)$$

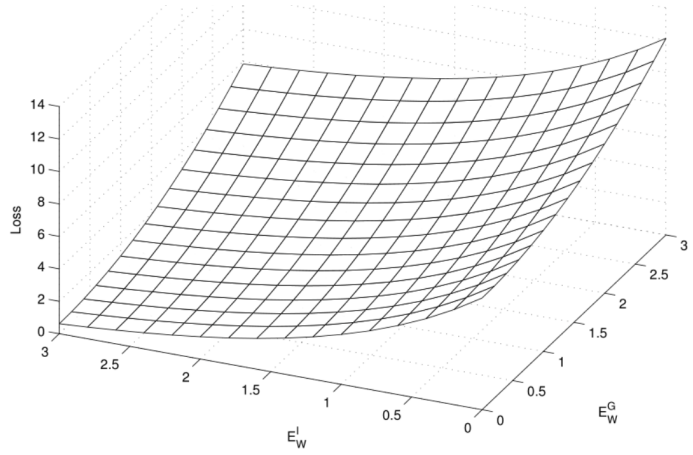

图 4: 损失函数 H 与 E_ω^G 和 E_ω^I 的三维图像

$$L(\omega,(Y,X_1,X_2)^i)=(1-Y)L_G(E_\omega(X_1,X_2)^i)+Y\cdot L_I(E_\omega(X_1,X_2)^i)$$

其中 $(Y,X_1,X_2)^i$ 是第 i 个样本，是由一对图片和一个标签组成的，L_G 是只计算相同类别对图片的损失函数，L_I 是只计算不同类别对图片的损失函数 P 是训练的样本数。通过这样分开设计，当最小化损失函数的时候，可以减少相同类别对的能量，增加不同类别对的能量。因此需要使 L_G 单调递增，使 L_I 单调递减。同时还需要满足不相同图片对的距离小于相同图片对的距离，即

$$E_\omega(X_1,X_2)+m<E_\omega(X_1,X_2')$$

则总的损失函数可以表示为

$$H(E_\omega^G,E_\omega^I)=L_G(E_\omega^G)+L_I(E_\omega^I)$$

2.3.4 优点

- 用一对样本的输入代替了单样本的输入，不再给单个样本确切的标签，而是通过给定一对样本是否属于同一类的 0、1 标签。
- 淡化标签，使网络具有很好的扩展性，可以对不属于训练集的测试集进行分类。
- 对小数量的数据集也使用，使得数据量较小的数据集也能够训练出不错的效果。

2.4　三元组损失函数 (Triplet Loss)

2.4.1　函数介绍及主要思想

在通常的监督学习过程中，由于有固定数目的分类，因此使用 softmax 函数和交叉熵（cross entropy）可以使网络较好地收敛。但是在人脸验证的问题当中，分类的数目并不重要，我们需要对比两张陌生的人脸（不属于训练集）并判断他们是否是同一个人。

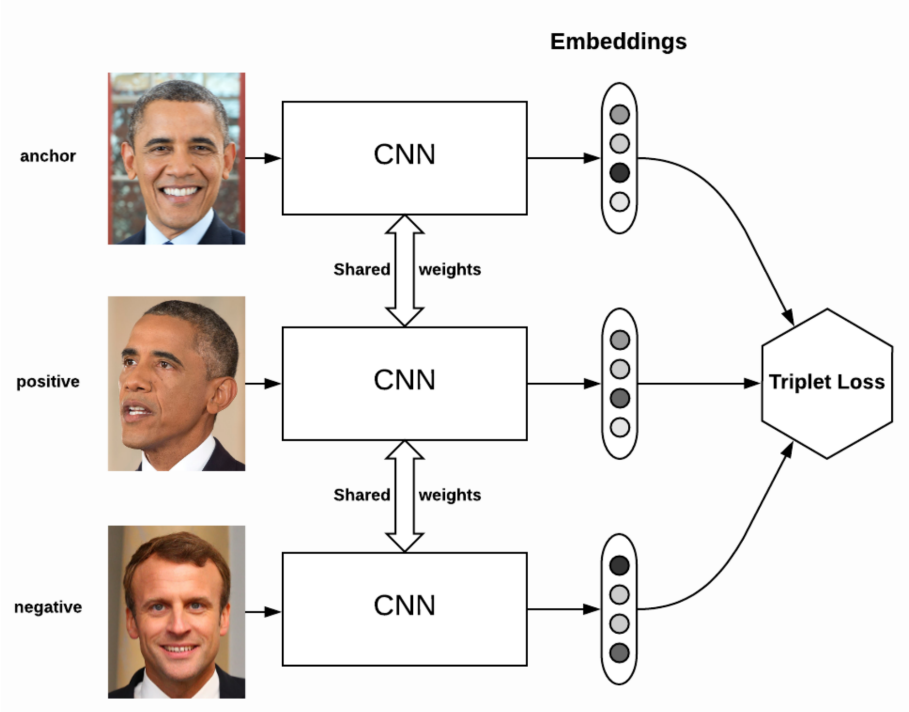

图 5: Triplet Loss 示例

在训练集中任意选取一张图片作为，锚点 (Anchor) 样本，选取与其属于同一类的一张图片作为正样本 (Positive)，再选取与其属于不同类的负样本 (Negative)。如上述示例中 Anchor 样本和 Positive 样本都是奥巴马，而 Negative 样本则是马克龙。[5]

训练的目标是最小化 Anchor 样本和 Positive 样本之间的距离，最大化 Anchor 样本和 Negative 样本之间的距离，如上图所示。因此我们可以设置损失函数为

$$\Gamma = max(d(a,p) - d(a,n) + margin, 0)$$

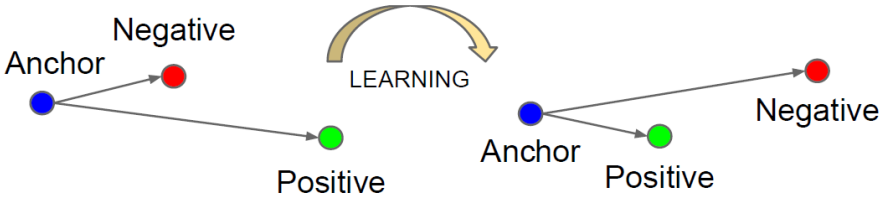

图 6: Anchor、Positive、Negative 三种样本的关系

其中 $d(a,p)$ 和 $d(a,n)$ 分别表示 Anchor、Positive 样本之间的距离，以及 Anchor、Negative 样本之间的距离。需要最小化损失函数即尽可能使得 $d(a,p)$ 接近 0，使得 $d(a,n)$ 大于 $d(a,p)+margin$，其中 margin 是人为设定的间距。[4]

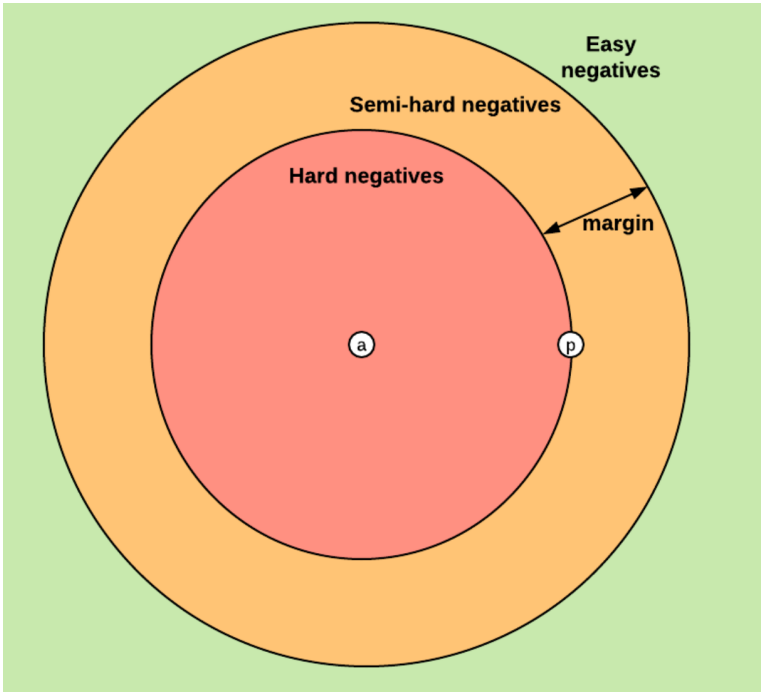

图 7: 给定 anchor 和 positive，三种情形下的 negative 范围

在实际操作中，我们会遇到三种 triplet 的情形，如图所示：

- easy triplets:$d(a,p)+margin<d(a,n)$
  已经实现目标不需要再调整了。

- semi-hard triplets:$d(a,p)<d(a,n)$
  距离目标很近了，需要微调

- hard triplets:$d(a,p) < d(a,n) < d(a,p) + margin$
  Anchor 和 Positive 距离很远，反而与 Negative 距离很近，这是我们主要调整的对象。

2.4.2 在线三元组构造 (Online Triplet Mining)

在线三元组构造 (Online Triplet Mining) 是相对于离线三元组构造 (Offline Triplet Mining) 而言的，Offline Triplet Mining 的意思是在每一代 (epoch) 开始之前就先找好 Anchor、Positive、Negative 的三元组，这样会花费大量的时间去生成 triplets，效率不是很高。Online Mining 则采用边训练边生成 triplets 的方法。

假设一个批次 (batch) 的人脸输入数目为 $B = PK$，其中 P 是人数，K 是每个人的人脸图像数目。有两种设 batch 的方法：

- batch all: 选取所有有效的 triplets，计算 hard triplets 和 semi-hard triplets 的损失平均值。这里我们不考虑 easy triplets，因为它们的损失是 0，如果选取会使得平均值大大降低。这样就产生了共 $PK(K-1)(PK-K)$ 个 triplets，其中包括 PK 个 anchor，每一个 anchor 对应 (k-1) 个 positive 和 (PK-K) 个 negative。

- batch hard: 对于每一个 anchor，选取当前 batch 中距离其最远的 positive 和距离其最近的 negative。这样就产生了 PK 个 triplets，它们是当前 batch 中每一个 anchor 对应的调整空间最大的 triplet。

实验表明，batch hard 调整策略效果最好。[3]

3 实现过程

3.1 数据处理

3.1.1 Siamese 网络数据处理

由于孪生网络的特殊性，每次将 2 个 input 放入同一个网络中，我们需要通过 LOSS 的计算来评价两个输入的相似性。那么，无论是 Train.txt 和 val.txt，我们都需要同时取两个 batch，将其作为 input1 和 input2 放入 network 中。一开始我们小组考

虑的是将整个 train.txt 读入后 shuffle 重新打乱后，每次连续取两个 batch 来进行训练，后来我们意识到这是不合适的。因为要训练一个好的 Siamese-network，其实需要**正样本**和**负样本**数目相对均衡，而从完全的 40 万数据重新 shuffle 后很难有来自同一个类别的照片同时输入一个网络，那么基本都是负样本，网络就会失去解决问题普遍性而变得更加倾向于将 negative_distance 加大化，即更倾向去输出-1，判断两张图片不来自同一个类。

所以我们需要对读入的数据进行 50% 正样本，50% 负样本的 shuffle。即

- 每两个 batch 为一批，每次通过 shuffle 生成 2 个 batch，然后循环直至完成对整个 train 数据的打乱。
- 每批中，有一半的数据为相同类，有一半的数据是不同类。
- 相同类数据的产生：随机出一个类别，再随机从该类别中选出两张图，放在两个 batch 的对应位置。
- 不同类则随机出两个类别, 再从两个类别分别随机选出两张图片。同时做查重和去漏的标记。
- 每一个 epoch 后再将数据重新按照 50%-50%shuffle, 保持每个 epoch 训练数据的不同。

（data 处理的代码已经以 function 的形式放在 util.py 中）

在这种处理下，我们可以保证所有的 433041 个 train 数据全部用上且保持 Positive_sample 和 negetive_sample 尽量均衡。训练出来的网络空间具有更好的区分度和鲁棒性。

验证集我们的处理方法也类似。将验证数据打乱并保持正负样本的一定比例性，多个 epoch 验证求得平均准确率，可以获得具有更高可靠性的 Accuracy。

3.1.2 Resnet 和 VGG 网络数据处理

因为 Resnet 和 VGG 处理的是一个分类问题，输出是图片的特征向量。每次输入一个 batch，那么对 batch 中的数据类别就没有特殊的要求。可以直接进行 shuffle 对 train 和 val 进行打乱，保持每个 batch 取到的数据的相异性。

参考文献

[1] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), volume 1, pages 539–546 vol. 1, June 2005.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. CoRR, abs/1512.03385, 2015.

[3] A. Hermans, L. Beyer, and B. Leibe. In defense of the triplet loss for person re-identification. CoRR, abs/1703.07737, 2017.

[4] O. Moindrot. Triplet loss and online triplet mining in tensorflow, Oct. 2018.

[5] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.