Non-I.I.D. Image Classification

郑凯文

2021 年 6 月 13 日

1 实验概述

实验使用NICO[3]数据集进行Non-I.I.D.的图像分类任务，数据集中共有10种Label和10种Context的图像（以512维特征向量表示），我们希望在只具有其中一些Context的样本的情况下，在其它Context下也能达到较高的预测准确率，即跨Context的泛化性能。

由于不同Label下的同一Context含义可能不同，这限制了方法的选取。根据提供的参考文献，我尝试了一些主流的变量解耦方法，通过对变量的加权，或对特征提取网络的正则化，使得进入分类器的变量之间线性无关。除此之外，我还尝试了使用简单的残差结构和注意力机制提取Context无关的特征。做出的创新有：

- 对于DWR和PFDL两种变量解耦方法，相对于原论文中的公式和实现，使用线性代数技巧将循环化为矩阵运算，从而大大提高运算效率和减小显存占用
- 提出促进不变特征提取的Variance loss，实践证明可以提高训练稳定性，减缓过拟合的发生

2 算法与模型

2.1 Baseline

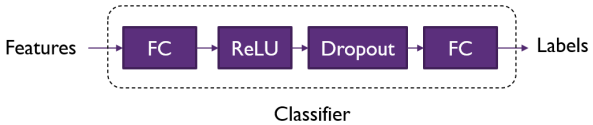

图 1: 分类器

图1的2层MLP是作为Baseline的分类器，使用了ReLU作为激活函数，并添加Dropout层（rate=0.3）以减缓过拟合。由于数据集中的样本已经是ResNet特征采样，因此可以将其直接接入分类器以得到标签。

训练时对线性层权重采用L1正则化，使用交叉熵作为损失，使用Adam优化器并应用了学习率衰减策略。经过对参数空间的搜索，设置L1正则化系数为1e-5，学习率为1e-2。

2.2 Decorrelated Weighting Regression[1]

设特征矩阵为$X_{n\times p}$，其中n为样本数量，p为特征长度即变量的数量。对于两个变量$X_{,j}, X_{,k}$，它们是线性独立的当且仅当对于任意环境a, b，都有$E[X^a_{,j}X^b_{,k}] = E[X^a_{,j}]E[X^b_{,k}]$。我们期望对样本施加权重$W_{n\times 1}$，使得加权后变量间线性无关。

设$\sum_{i=1}^{n} W_i = n$，则可以对损失添加上述期望差值的平方作为正则项

$$L_B = \sum_{j=1}^{p} ||X^T_{,j}\Sigma_W X_{,-j}/n - X^T_{,j}W/n \cdot X^T_{,-j}W/n||_2^2 \tag{1}$$

其中$X_{,-j}$是除第j个外的所有变量，实现中可以通过对特征矩阵的第j列置0得到。上式需要对特征的长度求和，实验中$p = 512$，这个过程耗时巨大，且由于损失是512个来源求和，计算图占用显存过多，不利于训练。为此，可以将上式转变为高效的矩阵运算：

上式中每个加数都是一个$1 \times p$大小的向量的二范数平方，我们可以将p个向量组合成一个$p \times p$的矩阵M，求M所有元素的平方和。下面将M表示为X和W的矩阵运算形式。为了表示的方便，将W归一化，这样就可以去掉上述损失计算中除以的n。

对于$M_{i,j}$，它是向量$X^T_{,j}\Sigma_W X_{,-j} - X^T_{,j}W \cdot X^T_{,-j}W$的第$i$个元素。经过简单的计算，其表达式为

$$M_{i,j} = \begin{cases} \sum_{k=1}^{n} W_k X_{k,i} X_{k,j} - \left(\sum_{k=1}^{n} W_k X_{k,i}\right)\left(\sum_{k=1}^{n} W_k X_{k,j}\right), & i \neq j \\ 0, & i = j \end{cases} \tag{2}$$

即

$$M = (X^T\Sigma_W X - (X^TW)(X^TW)^T) \cdot (1 - diag(1, 1, \ldots, 1)) \tag{3}$$

$$L_B = ||M||_2^2 \tag{4}$$

通过上述矩阵运算，DWR中的损失函数可以以很低的成本计算，这使得实现时得以采取full-batch的训练方式。

设网络参数为β，除了上述损失，还添加了L1正则化以及对W的正则化，总的损失为

$$L = WL_C + \lambda_1||\beta||_1 + \lambda_2 L_B + \lambda_3||W||_2^2 + \lambda_4(\sum_{i=1}^{n} W_i - 1)^2 \tag{5}$$

其中L_C为所有样本的交叉熵损失向量。实验中，设置$\lambda_1, \lambda_2, \lambda_3, \lambda_4$分别为1e-2，10，0.1，100。在训练时，每个Epoch分两步对样本权重W和网络参数β分别进行更新：

- 固定β，利用损失L训练W
- 固定W，利用损失L训练β

2.3 Deeper Network with Attention

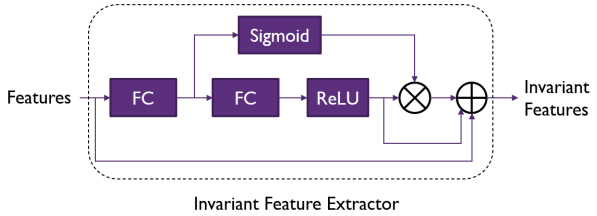

图 2: 不变特征提取网络

在Baseline的基础上，我尝试设置更加复杂的网络结构。利用注意力机制和残差结构，可以设计图2所示的网络。我使用它的目的是作为一个Context无关的不变特征提取器。虽然给定的已经是ResNet得到的特征，但还可以将其进一步变换来尽量与Context解耦。这样，输入特征经过上述网络得到不变特征，不变特征再经过分类器得到最终的输出标签。

2.4 Partial Feature Decorrelation Learning[2]

PFDL与DWR不同，它假设了一个特征提取网络f，希望f能提取出线性无关的特征$U = f(X)$。设稳定变量为S，不稳定变量为V，论文中假设S可以分解为$S = S_{ind} + \hat{g}(V)$，其中S_{ind}是与V无关的部分。于是可以训练一个特征分解网络g来近似$\hat{g}$，g在实现中只是一个简单的线性变换。

对于特征分解网络g，其损失为

$$L_g = \frac{1}{np}\sum_{i=1}^{n}\sum_{j=1}^{p}||U_{i,j} - \tilde{U}_{i,j}||_2^2 \tag{6}$$

其中$\tilde{U}_{i,j} = g(U_{i,-j})$。对于特征提取网络$f$，需要使$S$与$V$线性无关，一种做法是使$E[g(V)^T V] = 0$来使特征部分解耦。于是，可以添加如下的去相关损失作为正则项

$$L_{decorr} = \frac{1}{p}\sum_{j=1}^{p}||\frac{1}{n}\sum_{i=1}^{n}\tilde{U}_{i,j}^T U_{i,-j}||_2^2 \tag{7}$$

在实现时，上式同样可以将求和化为矩阵运算。对于特征提取网络f和分类器z，算法每个Epoch的训练流程为：

- 固定f, z，使用损失L_g优化g
- 使用分类损失+去相关损失作为总损失，固定g，优化f, z

其中f我使用的是上节的不变特征提取网络，z则是两层MLP。训练时同样使用了L1正则化。

2.5 Attention with Variance Loss

上节PFDL可以作为一种对更深网络的正则化方法，促进训练。而回归不变特征提取的初衷，可

以对中间得到的Invariant feature添加正则项：为了使提取的特征是Context无关的，希望尽量减小中间层Feature对于同一Label、不同Context的方差。

于是，定义如下的Variance loss：

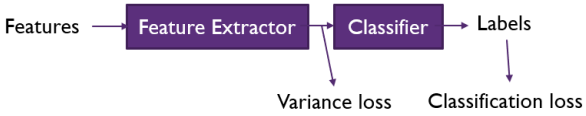

图 3: 为中间层特征添加Variance loss

$$L_{var} = \sum_{k=1}^{K} D[U_k] \tag{8}$$

其中K为类别数，U_k为标签为k的训练样本提取得到的中间层特征。如图3所示，Variance loss作为一个正则化项，约束特征提取网络，使其类似于无监督预训练，与之后分类器的训练分离。这相当于让网络自己去学习哪些是“稳定变量”，并使用它们从同一类别的样本提取出类似的特征。

3 实验结果与分析

3.1 验证集准确率

按照训练集：验证集=6：1划分，其中验证集使用完全不同的Context。训练1000个Epoch，取验证集最高准确率，结果如下：

表 1: 验证集准确率

方法	准确率
Baseline	76.7%
DWR	77.5%
Attention	76.2%
PFDL	77.5%
Attention with Variance Loss	77.7%

其中Baseline相较于汇报时有所提高，是因为额外使用了L1正则化并进行了更细致的调参。与这个Baseline对比的方法均类似地使用了L1正则化。

从表中看出，DWR作用于2层MLP，可以将验证集准确率提升1%左右。而额外添加特征提取网络后，Attention的准确率反而下降不少。从训练过程可以看出，训练集准确率最后维持在99%以上接近100%，说明加深网络后，训练进行良好，但发生了十分严重的过拟合：Baseline最终的训练集准确率也接近100%，但过拟合现象没有它严重。而使用PFDL和Variance loss后，过拟合现象得到了一定的缓解，验证集准确率也有1%左右的提升。

3.2 Variance Loss对于过拟合的作用

在这个Non-I.I.D.分类问题中，最突出且易于观察的现象是过拟合。这可以从训练集准确率和验证集准确率的差异进行观察：Baseline和Attention的训练集准确率都接近100%，使用DWR、PDFL后，训练集准确率在98%左右。而Variance loss对于过拟合的缓解最为显著。

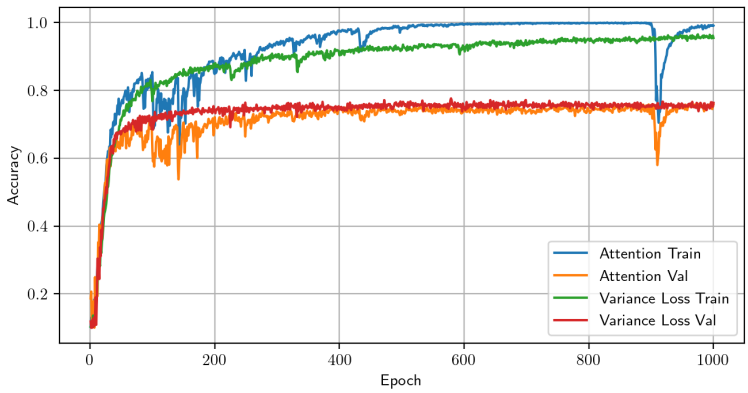

图 4: 添加Variance loss前后的过拟合现象

图4绘出了添加Variance loss前后，Attention模型的学习曲线。Variance loss对于缓解过拟合、增强训练稳定性具有很大作用：

- 添加Variance loss前，训练集错误率接近100%，而添加后在95%左右。
- 从训练稳定性来看，添加Variance loss后，随着训练的不断进行，验证集准确率维持在77%左右，十分平稳；而不添加时，出现了许多抖动。

3.3 不变特征提取可视化

将Attention with Variance Loss模型中特征提取网络的输出与输入特征可视化，可以检验Invariant Features提取的效果。

图5是使用t-SNE将512维特征嵌入到二维平面的效果，其中o代表输入特征，x代表特征提取后的特征，同一颜色代表同一类别。出乎意料的是，特征提取网络并没有使得同一个Label样本的聚集情况得到较大提升。这应该是由于特征提取网络不够复杂，表达能力不足：我尝试过只使用Variance loss而不使用分类损失，开始时Variance loss确实有明显下降，但训练约200个Epoch后便很难下降了，这说明特征提取网络的非线性性不足以将同一Label的样本映射到相似的不变特征。即便如此，Variance loss仍对减缓过拟合有一定的帮助。

4 总结与反思

虽然在实验过程中，验证集准确率有着不错的数值，但最终的测试集准确率只有74.5%，十分令人不满意。我也对预测文件提交的失误进行了反思：虽然尝试的方法很多，但我们的实验在数据

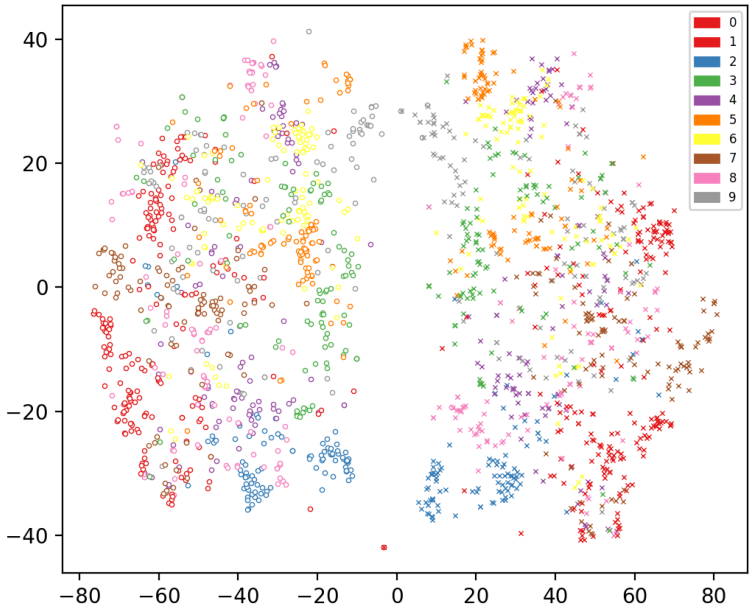

图 5: Invariant Features

集划分上进行得并不充分，只随便选了一个6：1的划分便在其上做完了所有实验。因此测试集上的失利也是可以预料的：

- 无论哪些改进，对于Baseline的提升也只有1%左右
- 训练集的划分是否过于片面，比如选择的训练集恰好和测试集差异较大
- 在验证集上准确率高，并不一定在测试集上表现好

课上其他同学的报告也给了我很多启发，表现好的方法可能只用了非常简单的模型，问题的根本在于解决过拟合上。比如设置高达95%的drop rate，比如使用上测试集，添加将训练集和测试集区分开的简单分类器（这类似于NLP中的伪标签方法，似乎是一种用上了无标签数据的半监督学习）。而最基本的解决数据集划分片面性的方法就是集成学习。

4.1 稳定性测试

使用模型Attention with Variance Loss，我额外进行了稳定性测试以检验上面的第2、3条，方法是将数据集划分为训练集：验证集：测试集1：测试集2=4：1：1：1，取验证集上准确率最高的模型，在两个测试集上测试。

表 2: 验证集准确率

数据集	准确率
训练集	93.5%
验证集	75.0%
测试集1	79.9%
测试集2	73.1%

可以发现，虽然总体准确率都在70%到80%之间，但验证集可以和测试集相差非常大，这也不难解释为什么我们提交的预测文件准确率比验证集低了3%。因此，如果进行一次训练集/验证集的划分，随机性是非常大的，在测试集上的表现不可预知，运气不好可能就是倒数的水平，毕竟本身也相差不大。

要解决这一问题，最简单的办法就是集成学习了，进行7折交叉验证，将7个模型的预测进行多数投票，这样就避免了数据集利用的不充分和单个模型的过大偏差。

总而言之，这是我第一次接触Non-I.I.D.图像分类，因此由于经验不足犯下了一些失误。从这个经历中我了解到，Non-I.I.D.很本质的一个问题就是解决过拟合，非常有效的方法是集成学习、高drop rate，还可以利用上测试集数据进行训练。模型可以很简单，方法可以很极端，但都是为了解决过拟合服务。当然这只是一种思路，另一种方法应该是使用复杂的网络进行对抗学习、构建生成模型等，但需要较多的训练数据。在样本只有几千时，简单模型或许更胜一筹。

参考文献

[1] Kun Kuang, Ruoxuan Xiong, Peng Cui, Susan Athey, and Bo Li. Stable prediction with model misspecification and agnostic distribution shift. In AAAI, pages 4485–4492, 2020.

[2] Yu Z, Wang P, Xiang C, et al. Partial Feature Decorrelation for Non-I.I.D Image classification[J]. 2020.

[3] Yue He, Zheyan Shen, Peng Cui. Towards Non-I.I.D. Image Classification: A Dataset and Baselines. Pattern Recognition, 2020