机器学习大作业

162050127 颜劭铭

2022 年 6 月 12 日

摘要

偏标记学习中的图像被模糊标注了多个候选标签，而其中只有一个是正确的标签，并且在训练阶段无法访问该标签。我们的目标是从提供的训练集中训练出一个正确的分类器，能够对于测试示例做出正确的预测。但是传统的消歧方法如基于平均和基于辨识的消歧方法随着数据量的增加效果会越来越差。在本文中，作者为了解决传统方法表示能力和辨识能力的不足，提出了新的方法“Deep Discriminative CNN”，利用深度卷积神经网络对于模糊标记的图像进行分类，可以提取到图像的深层信息，增加表示能力。在此基础上，加入了一个基于熵的正则化器和利用不同时期的时间组合预测来指导训练过程，可以更好地突出候选标签集中的潜在真实标签，增强辨识能力。作者将他们的方法与其他现有的最先进方法相比，在多个数据集上的实验结果证明了他们提出的 D^2CNN 的有效性.

关键字：偏标记学习，D^2CNN，时间集成，熵

1 论文相关背景介绍及所要解决的问题

1.1 偏标记学习（PLL）

在传统的监督学习里面，我们可以将训练集的标签称为监督信息。基于标签的质量我们可以将监督学习分为强监督学习和弱监督学习，其中强监督学习的标签一般是明确且唯一的，而弱监督学习中包含了如半监督，多标签，多示例，偏标记。

在很多场景中，我们对于一张图片的标记通常采用众包进行，而不同水平的标注者所打上的标签是不同的，这就是我们这张图片的候选标签集，那么我们所需要的应该是该图片的真实标签，我们该如何从候选标签集中选出真实标签呢？

偏标记学习中令 $\mathcal{X} \in \mathbb{R}^d$ 表示 d 维输入空间，y 表示 c 类的标签空间，因此可以将训练集表示为 $\mathcal{D} = \{(x_i, S_i) \mid 1 \leqslant i \leqslant n\}$，其中 x_i 表示第 i 张图片，S_i 指的是第 i 张图片的候选标签集，可以表示为 $S_i = A_i \cup \{y_i\}$，A_i 表示第 i 个图片中错误的标签集，而 y_i 指的是 ground truth，因此偏标记学习的目标就是从训练集 $\mathcal{D}$ 中训练一个分类器可以从候选标签集中选出 ground truth。

在偏标记学习框架下, 每个样本的标签信息不再具有单一性和明确性, 真实标签隐含在候选标签集中. 针对这种具有歧义性的样本, 一种直观的解决偏标记学习问题的方法是对候选标签集进行消歧，主要用基于平均的消歧策略和基于辨识的消歧策略两种方法。

基于辨识的消歧将偏标签样本的真实标签作为隐变量, 通过迭代的方式优化内嵌隐变量的目标函数以实现消歧，主要使用最大似然准则和最大边际准则两种学习策略。但是该方法有一个潜在缺点是当前迭代步骤中识别的标签可能会被证明是错误的，并且在后续的迭代中很难纠正。

基于平均的消歧赋予偏标签样本的各个候选标签相同的权重, 通过平均各个标签在模型上的输出以进行预测 ground truth。该方法有个缺点是由于候选标签中的错误标签数量是远远多于真实标签的，因此错误标签的输出可能会压制真实标签的输出，大幅度降低性能。

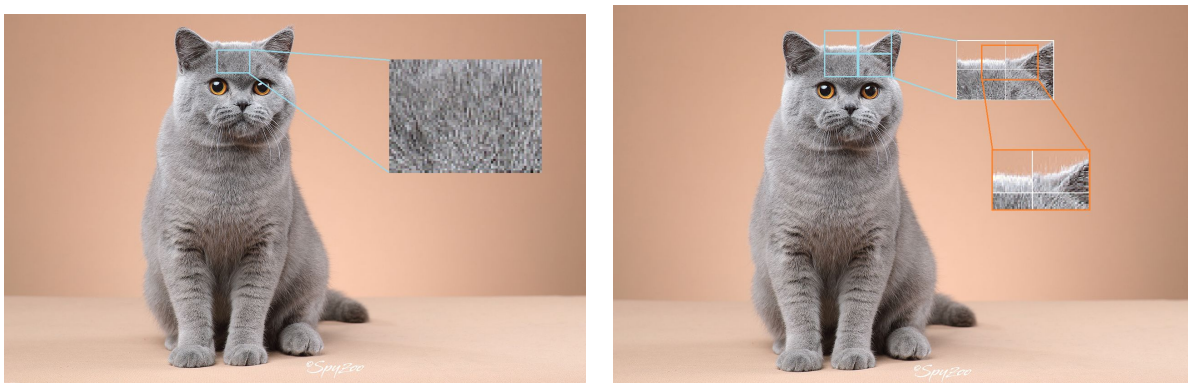
随着候选标签集的增大，错误标签引入的负面影响会越来越大。除此以外，在作者发表论文之前，偏标记学习中的图像分类问题都是使用非深度网络进行的，因此他们使用的特征都是传统手工特征，只包括底层和中间特征。

基于以上所述，现有方法通常缺乏表示能力和辨别能力，前者是由于浅层学习框架造成的，后者是由于消歧技术不完善造成的。

1.2 深度卷积神经网络（DCNN）

在前文中有提到传统的偏标记学习通常采用非深度神经网络框架进行，学习到的通常是传统手工特征，可以提取到的图像信息非常有限，所以准确度有待提高。为了解决这个问题，作者创新性地第一次采用了深度卷积神经网络框架，DCNN 通常由卷积层，池化层和全连接层构成，在前几层可以捕获层次较低的特征，而随着训练层数的加深，可以挖掘图像更深、更为抽象的特征。可以用以下的例子进行解释。

在卷积神经网络每一个卷积核在单层中只能计算相邻区域的像素点的响应，所以，在第一层中，我们的网络只能计算出每一个小的像素区域的响应值，比如，第一层的某个卷积核就只能计算下图中蓝框框出来的区域中的所有像素的响应值：



(a) 第一层卷积　　(b) 第二层卷积

图 1: 卷积示意图

这片单个卷积核在原图上能够计算的响应区域被称为该卷积的感受野。我们可以发现这个小感受野所提取的信息是非常少的，基于蓝色框中的信息我们可能会把它判断为沙发，石头，不一定能判断为蓝色英短。因此如此小的感受野是无法提取到高阶的图像语义信息，所以我们进行第二层的卷积。

我们可以发现到第二层的时候感受野变大了，能够计算更大区域像素点的响应。并且由于第一层卷积核还会对于所提取区域的色彩，纹理模式等信息进行计算，因此到第二层卷积层时，卷积核还得到了颜色，纹理等编码信息，我们得到了更大的特征空间。

因此随着训练层数的加深，卷积核的感受野逐渐变大，更有可能提取到深层信息，也就是图像的高阶语义信息，并且它积累的特征空间更大，更有把握去判断这张图片是一只英短。

2 解决问题的思路与方法

首先在上文中有讲述到偏标记学习中对于输入图片和对应标签的表示形式。在本篇论文中，作者重新令 $X = [x_1, \cdots, x_n]$ 表示训练集，x_i 表示第 i 张图片，用列向量 y_i 表示 c 维的候选标签集，因此作者用一个 $n \times c$ 的模糊标签矩阵 $Y = [y_1, \cdots, y_n]^T \in 0, 1^{n \times c}$ 来表示每张图片及它的候选标签集，其中 $y_{ij} = 1$ 表示第 j 个标签是第 i 个图像 x_I 的候选标签，否则 $y_{ij} = 0$。

2.1 框架

在前文中有提到作者采用了深度卷积神经网络作为该任务的框架，尽可能地提取图像的高阶语义信息，以提高模型的表示能力。而由于 ResNet 在各种计算机视觉任务中取得了非常令人印象深刻的表现，所以作者最终采用了 ResNet。

ResNet 在 2015 年由何恺明等人在 Deep Residual Learning for Image Recognition [1] 中提出。在深度学习的发展过程中，人们产生了疑问：层数越多，训练效果一定越好吗？如何优化过深的神经网络？在经过试验后发现 56 层的神经网络甚至比 20 层的神经网络效果更差，也就是说新增加的 36 层是对神经网络的“恶化”。那么，如果这 36 层神经网络是恒等映射，那么 56 层的神经网络不就和 20 层的一样好了吗？更进一步的话如果这 36 层神经网络相比于恒等映射再好上那么一点点（更接近最优函数），那么就起到了正优化的作用了。

因此假设某一层内，最优函数记为 $H(x)$，那么我们所拟合的目标函数 $F(x)$ 定义为 $F(x) = H(x) - x$，函数 $F(x)$ 被称为“残差函数”。这样的方法是基于假设：最优函数与线性函数有较高的相似性。ResNet 的基本框架可由如图 2的残差快表示：

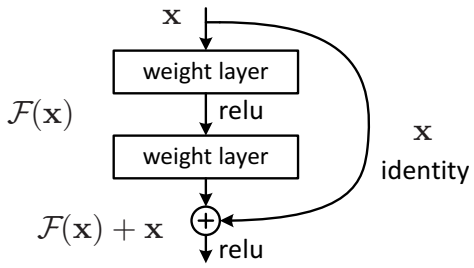


图 2: 残差块

对应到神经网络中，残差快的数学表达式可以写成 $y = \sigma(F(x, W) + x)$，其中，y 代表残差块的输出，$\sigma(\cdot)$ 代表激活函数，$F(\cdot)$ 代表残差函数，x 代表输入，W 代表残差块内的所有权重。

2.2 损失函数

交叉熵是一种广泛应用于传统的监督分类任务，为我们提供了一种表达两种概率分布的差异的方法。当 p 和 q 的分布越不相同，p 相对于 q 的交叉熵将越大于 p 的熵，形式上可以被定义为：

$$H_p(q) = \sum_x q(x) \log_2 \left(\frac{1}{p(x)} \right) = -\sum_x q(x) \log_2 p(x) \tag{1}$$

但是如果直接最小化原始的模糊标签和预测标签之间的交叉熵意味着所有的候选标签都被平等对待，因此可以看作是一种基于平均的消歧方法，而这种消歧方法的缺点在上文已经分析过了。因此需要对于交叉熵进行改进以应用到该任务中。

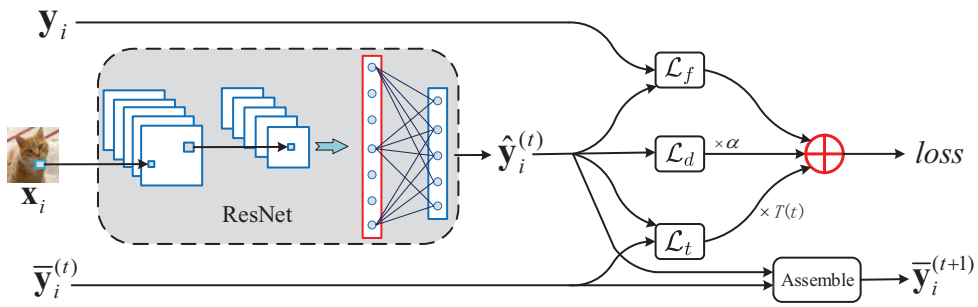


图 3: 论文方法的 pipeline

为了更好地完成消歧的目标，作者定义了如下所示的损失函数：

$$\text{Loss} = \mathcal{L}_f(\mathbf{Y}, \hat{\mathbf{Y}}) + \alpha \mathcal{L}_d(\hat{\mathbf{Y}}) + T(t) \cdot \mathcal{L}_t(\overline{\mathbf{Y}}, \hat{\mathbf{Y}}) \tag{2}$$

结合图 3和公式，图中的 y_i 指的是原始的模糊标签，x_i 指的是输入的图像，$\overline{\mathbf{y_i}}^{(t)}$ 指的是前一轮的训练目标矩阵，等式右边的第一项 $\mathcal{L}_f$ 被称为保真项，计算模型网络预测结果的 $\hat{\mathbf{y}}_\mathbf{i}$ 和原始的模糊标签 $\mathbf{y_i}$ 之间的损失；第二项 $\mathcal{L}_d$ 被称为判别项，通过最小化熵来降低标签的不确定性；第三项 $\mathcal{L}_t$ 被称为时间集成项，使前面阶段的训练结果与当前步骤的预测达成一致。

2.2.1 保真项

保真项的公式定义如下：

$$\mathcal{L}_f(\mathbf{Y}, \hat{\mathbf{Y}}) = -\frac{1}{n}\sum_{i=1}^{n}(\mathbf{1}-\mathbf{y}_i)^\top \log(\mathbf{1}-\hat{\mathbf{y}}_i) \tag{3}$$

我们可以认识到，ground truth 一定存在于该图像的候选标签集中。因此对于预测标签向量中不属于该图片的候选标签对应的元素应该为 0。因此为了保证 ground truth 一定来自于候选标签集，作者提出了上述公式，其中 1 指的是一个全 1 向量，通过最小化全 1 向量与原始模糊标签和预测标签相减的交叉熵，可以使非候选标签集中的标签是 ground truth 的概率被约束为零。因此，预测标签只能来自候选标签集。

2.2.2 判别项

判别项的公式定义如下：

$$\mathcal{L}_d(\hat{\mathbf{Y}}) = -\frac{1}{n}\sum_{i=1}^{n}\hat{\mathbf{y}}_i^\top \log \hat{\mathbf{y}}_i \tag{4}$$

我们知道熵可以用来表示一个标签的不确定性，并且当熵减小的时候，不确定性也随之减小。同时我们可以知道预测标签的输出形式 $\hat{\mathbf{y}}_\mathbf{i}$ 应该满足 $\sum_{j=1}^{c}\hat{y}_{ij}=1$ 并且 $\hat{y}_{ij}\geqslant 0$，因为在理想情况下预测标签将最有可能是正确标签的第 j 个标签的元素输出为 1，其他输出为 0。因此最小化 $\hat{y}_i$ 的熵将扩大可能标签和不太可能标签之间的值差距，使得预测更加准确。

我们可以用图 4的例子和坐标轴来理解：

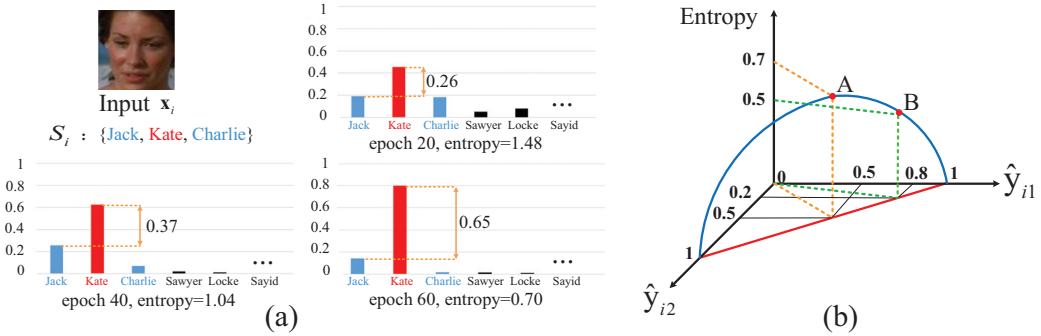


图 4: 判别项例子

图 4.(a) 中显示了一张人脸图像，其候选标签集包含“Jack”、“Kate”和“Charlie”，其中“Kate”是 ground truth 标签。随着训练的进行，标签向量中最大的标签输出和第二大的标签输出之间的边距增加，这表明标签向量的不确定性降低了。因此，标签向量的熵也会下降。最终，我们可以得到预测标签为“Kate”。

图 4.(b) 反映对于一个二分类问题他的熵的变化过程，其中红线是可行区域，蓝色曲线记录了位于红线上的每个预测标签向量的熵。当 $\hat{\mathbf{y}}_\mathbf{i}=[0.5,0.5]^T$，熵达到最大值，这是最模糊的预测向量。因此，最小化熵将会强制 $\hat{\mathbf{y}}_\mathbf{i}$ 逼近预测向量，如 $[0,1]^T,[1,0]^T$。

2.2.3 时间集成项

首先介绍一种生成组装的训练目标矩阵的方法，滑动平均（指数加权平均）生成方法：

$$\begin{aligned}\mathbf{S}^{(t)} &= \gamma\mathbf{S}^{(t-1)} + (1-\gamma)\hat{\mathbf{Y}}^{(t)} \\ \overline{\mathbf{Y}}^{(t+1)} &= \mathbf{S}^{(t)} / \left(1-\gamma^{t}\right)\end{aligned} \tag{5}$$

指数加权平均是一种近似求平均的方法，$\mathbf{S}^{(t)}$ 指的是最近的 $\frac{1}{1-\gamma}$ 次训练目标的矩阵的平均值，$\hat{\mathbf{Y}}^{(t)}$ 指的是第 t 次训练目标矩阵，β 是一个超参数，$\beta = 0.9$ 时，$\mathbf{S}^{(t)}$ 与过去 10 次结果相关。

对于上面第一个公式进行展开我们会发现 $\mathbf{S}^{(t)}$ 实际上是对于每次训练结果的加权平均，时间越近，权重越大，而且是指数形式的，因此称为指数加权平均。

第二个公式是为了纠正 $\mathbf{S}^{(t)}$ 的启动偏差，可以得到下一次的训练目标 $\hat{\mathbf{Y}}^{(t+1)}$。

作者受到半监督学习的时间集成技术 [2] 的启发，给出时间集成项的公式定义如下：

$$\mathcal{L}_t(\overline{\mathbf{Y}}, \hat{\mathbf{Y}}) = -\frac{1}{n}\sum_{i=1}^{n} w_i \cdot \overline{\mathbf{y}}_i^{\top} \log \hat{\mathbf{y}}_i \tag{6}$$

在上式中我们可以很容易理解，w_i 决定了 $\overline{\mathbf{y}}_i$ 应该在多大程度上被信任来指导整个网络的训练过程，当 $\overline{\mathbf{y}}_i$ 可能是 x_i 的真实标签向量时，w_i 应该比较大，所以应该更加关注。那么如何确定 w_i 呢？

作者将 $m_i \in [-1, 1]$ 表示为来自候选标签的 $\overline{\mathbf{y}}_i$ 的最大值与来自非候选标签的最大值之间的边距，即 $m_i = \max_{y_j \in S_i} \overline{\mathbf{y}}_{ij} - \max_{y_k \in S_i} \overline{\mathbf{y}}_{ik}$。那么，权重 wi 可以定义为：

$$w_i = \begin{cases} 0 & m_i \leq 0 \\ m_i^2 & \text{otherwise} \end{cases} \tag{7}$$

也就是说当 $\overline{\mathbf{y}}_{ij} < \overline{\mathbf{y}}_{ik}$ 的时候，$m_i < 0$，证明了 $\overline{\mathbf{y}}_i$ 是错误的，因此此时的权重为 0，我们应该不关注，而当 $m_i > 0$ 时，此时训练目标更可能是正确的，我们更应该关注它。作者通过测量每个训练目标的可靠性以决定模型依赖训练目标的程度，从而有利于模型精确地消除模糊标记的图像的歧义。

最后，我们回到作者给出的损失函数，发现还有一个时间加权函数 T(t)。当刚开始训练的时候，网络还处于一个非常初始的阶段，预测的结果也不是很准确，因此我们应该减少对于这部分训练目标的关注，而随着训练轮数的增加，网络的消歧能力增强，因此我们应该更关注后续的一些训练结果，因此作者给出了 T(t) 的定义如下：

$$T(t) = T_{max} \exp[-5(1-t)^2] \tag{8}$$

其中 T_{max} 指的是 T(t) 能达到的最大值，而 t 以线性的速度从 0 增长到 1，当 t 增大的时候，T(t) 也随之增大。

因此，通过引入时间加权函数，网络在初始训练阶段主要从原始的模糊标签中学习，并在训练过程进行时逐渐从组装的预测中学习，越来越关注网络的训练目标。

2.3 优化

作者在优化中主要采用了 BP 反向传播对于网络中的参数利用链式法则进行更新，以及 Adam 优化方法和指数加权平均对于网络中的参数 Θ 进行更新。

3 实验思路及改进

3.1 使用数据集介绍

为了更好地评估性能，作者采用了两个合成数据集 FM 和 SVHN 和两个真实数据集 Lost 和 Yahoo!News 进行对比实验，数据集的信息如图 5所示，其中带有后缀“-v1”的数据集意味着为每个包含的图像在候选标签集中添加一个额外的噪声标签，而后缀“-v3”意味着将另外三个不正确的标签合并到候选标签集中，“Avg #labels”表示单个图像的候选标签的平均数量。

Datasets	# Images	# Classes	Avg # labels
FM-v1	70,000	10	2
FM-v3	70,000	10	4
SVHN-v1	99,289	10	2
SVHN-v3	99,289	10	4
Lost	1,122	16	2.23
Yahoo!News	14,322	38	1.44

图 5: 数据集介绍

3.2 与其他非深度框架对比

从前文的分析中我们可以得知在以往的偏标记学习图像分类任务中通常采用非深度框架，因此作者对于不同数据集将自己提出的 D^2CNN 与其他非深度框架和 DCNN 进行了对比实验，其中 DCNN 采用的是交叉熵 (基于平均的消歧方法)，并用五折交叉验证得到各个框架的平均分类准确度，结果如图 6所示。

	FM-v1	FM-v3	SVHN-v1	SVHN-v3	Lost	Yahoo!News
RegISL	-	-	-	-	0.761 ± 0.037 •	0.598 ± 0.016 •
SURE	-	-	-	-	0.794 ± 0.037 •	0.729 ± 0.010 •
WMCar-ICE	-	-	-	-	0.795 ± 0.020 •	0.705 ± 0.010 •
MCar	0.913 ± 0.003 •	0.825 ± 0.002 •	0.796 ± 0.004 •	0.545 ± 0.004 •	0.743 ± 0.011 •	0.671 ± 0.010 •
PLKNN	0.897 ± 0.004 •	0.848 ± 0.004 •	0.736 ± 0.003 •	0.636 ± 0.002 •	0.651 ± 0.012 •	0.562 ± 0.017 •
M3PL	0.884 ± 0.002 •	0.874 ± 0.004 •	0.827 ± 0.002 •	0.788 ± 0.003 •	0.678 ± 0.032 •	0.613 ± 0.001 •
IPAL	0.912 ± 0.005 •	0.905 ± 0.003 •	0.798 ± 0.003 •	0.777 ± 0.001 •	0.790 ± 0.034 •	0.647 ± 0.017 •
DCNN	0.902 ± 0.007 •	0.890 ± 0.008 •	0.922 ± 0.003 •	0.908 ± 0.007 •	0.580 ± 0.031 •	0.740 ± 0.006 •
D^2CNN	**0.936 ± 0.002**	**0.927 ± 0.003**	**0.937 ± 0.003**	**0.929 ± 0.001**	**0.838 ± 0.014**	**0.833 ± 0.009**

图 6: 与其他框架对比结果

3.2.1 合成数据集

我们可以发现 D^2CNN 在这些合成数据集上实现了优于其他框架的性能。此外，我们可以清楚地观察到，当候选标签的数量增加时，所有框架的性能都会下降，尤其是在 SVHN 数据集上。然而，D^2CNN 的性能下降远小于其他基线，这进一步证明了所提出的 D^2CNN 方法的有效性和鲁棒性。

3.2.2 真实数据集

从图 6中我们可以发现 D^2CNN 在 Lost 数据集上的准确率比 IPAL、SURE 和 WMCar-ICE 高约 4%。在 Yahoo!News 数据集上，D^2CNN 显着优于其他框架，并以 9.3% 的优势领先于第二好的方法。

但是我们可以观察到一个有趣的现象，传统 DCNN 的性能在合成数据集上令人满意，但在实际数据集上比 D^2CNN 差得多。原因是具有交叉熵损失的 DCNN 可以看作是一种基于平均的消歧策略。当训练图像不足且复杂时，每个训练图像的唯一真实标签可能会被候选标签集中的错误标签所淹没，从而导致 DCNN 的性能有限。

3.3 方法的鲁棒性和有效性证明

3.3.1 鲁棒性

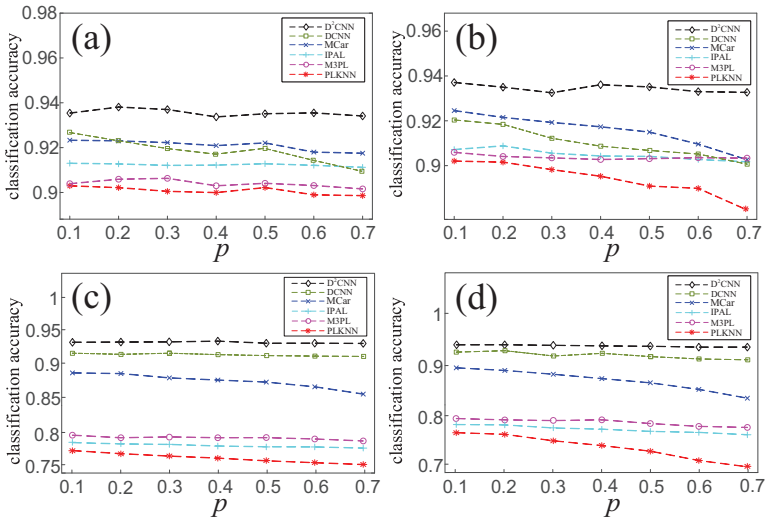


图 7: 鲁棒性

作者改变了合成数据集 FM 和 SVHN 中模糊标签图像占整个数据集的比例，并在候选标签分别为 1 和 3 时，对于不同的框架进行了对比实验，观察分类准确度，可以发现 D^2CNN 在数据集中不同比例的模糊标记图像下通常可以很好地工作，如图 7所示。

3.3.2 有效性

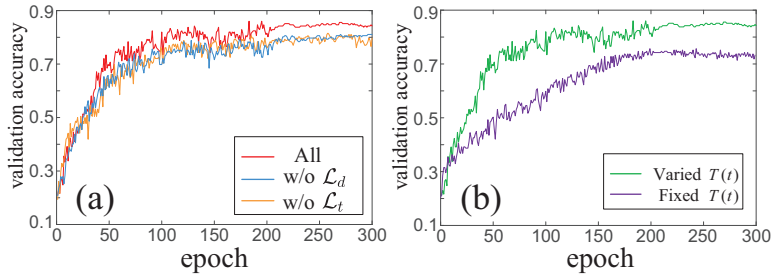


图 8: 有效性

作者对于提出的三个损失的有效性进行验证。

首先对于 $\mathcal{L}_f$ 和 $\mathcal{L}_d$ 进行消融实验验证，分别移除 $\mathcal{L}_f$ 和 $\mathcal{L}_d$ 以后观察 D^2CNN 的准确度，可以发现移除以后，准确度会有一定的下降，如图 8.(a) 所示，因此证明了 $\mathcal{L}_f$ 和 $\mathcal{L}_d$ 的有效性。

接下来对于时间加权函数 T(t) 进行验证，由于 T(t) 是一个随着 t 增加不断增加的量，因此作者将 T(t) 固定不变以进行对比实验，观察性能，如图 8所示，可以发现固定 T(t) 后 D^2CNN 的分类准确度下降了非常多。

3.3.3 学习图像深层特征提高表示能力的有效性

Methods	GIST Feature	Deep Feature
RegISL	0.761 ± 0.037 •	0.786 ± 0.025 •
SURE	0.794 ± 0.037 •	0.826 ± 0.006
WMCar-ICE	0.795 ± 0.020 •	0.798 ± 0.035 •
MCar	0.743 ± 0.011 •	0.811 ± 0.038 •
PLKNN	0.651 ± 0.012 •	0.705 ± 0.017 •
M3PL	0.678 ± 0.032 •	0.774 ± 0.032 •
IPAL	0.790 ± 0.034 •	0.824 ± 0.021 •
D^2CNN	-	**0.838 ± 0.014**

图 9: 图像深层特征和传统手工特征对比

在之前的实验中，处理模糊标签的图像分类任务的传统方法都是非深度的，仅适用于手工制作的特征。为了公平比较，作者还利用图像的深层特征作为其它框架的输入，并将它们的输出与我们在 Lost 数据集上的 D^2CNN 进行比较，如图 9所示。

我们可以发现图像深层特征在所有框架上的效果始终优于 GIST 特征，因此采用深度网络有利于提高性能；并且与其他采用深层特征作为输入的框架相比，作者提出的 D^2CNN 仍然具有最高的准确度，这变相地证明了作者提出的损失函数的有效性。

3.4 对于作者进行实验的想法

我觉得作者进行的实验是非常完善的，他不仅仅是将自己提出的框架 D^2CNN 与其他框架在相同数据集上进行实验证明该框架的性能优异，更是对于自己提出的损失函数进行了有效性的验证，以及对于图像深度特征和表层特征作为输入的实验效果进行比较，证明了自己提出的方法在表示能力和辨识能力方面的提高。

4 问题与改进

作者提出的损失函数是通过引入熵的判别正则化项和时间集成的正则化项来降低候选标签的不确定性，提高选中 ground truth 的概率，避免了基于平均和基于辨识的消歧策略的缺点。但是近年来又有了新的消歧策略——基于流形假设的消歧策略。

该消歧策略利用了流形假设认为相似的样本向量拥有相似的输出这一特性，尽可能从偏标记数据集中挖掘有用的信息。由于从标记来看，我觉得在图像分类任务中，一个图像的候选标签集中的候选标签是非常相似的，比如说文中提到的众包任务中对于 ground truth 为骡子的图像，错误的标签为马和驴，而流形假设正是基于输入相近的样本具有相同的输出，所以我觉得这种新的消歧策略可能会对于该任务带来一定的改进。

因此如果进一步进行完善的话，我觉得作者的实验过程中可以再加入一些利用该消歧策略的框架进行对比，甚至有可能该策略在某些数据集上会比起现在的框架更加有效。

参考文献

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[2] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016.