南京航空航天大學
Nanjing University of Aeronautics and Astronautics

模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

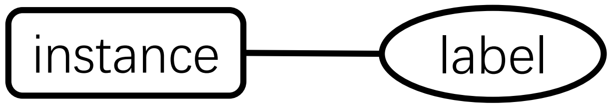
Exploiting Class Activation Value For Partial-Label Learning

ICLR 2022

CAM(Class Activation Map) -> **CAV**(Class Activation Value) -> **CAVL**

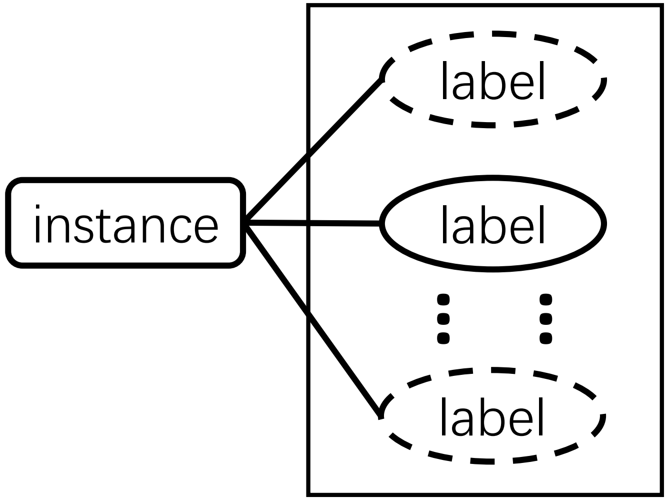
Problem to be solved : Partial Label Learning

- Ordinary supervised learning vs. **Partial label learning**



Ordinary supervised learning
(only one ground-truth label)

candidate label set



Partial label learning
(a candidate label set, among which only one label is valid)

Related Work :

1. **平均消歧策略** average-based strategy (ABS)

   平等地对待所有的候选标签

2. **辨识消歧策略** identification-based strategy (IBS)

   致力于不断地从候选标记集合中识别出最有可能的真实标记

PRODEN 假设真实标记在候选标记集中实现最小损失;

MGPLL 基于非随机标记噪声假设;

在这些方法中，只有少部分可以和深度神经网络兼容。

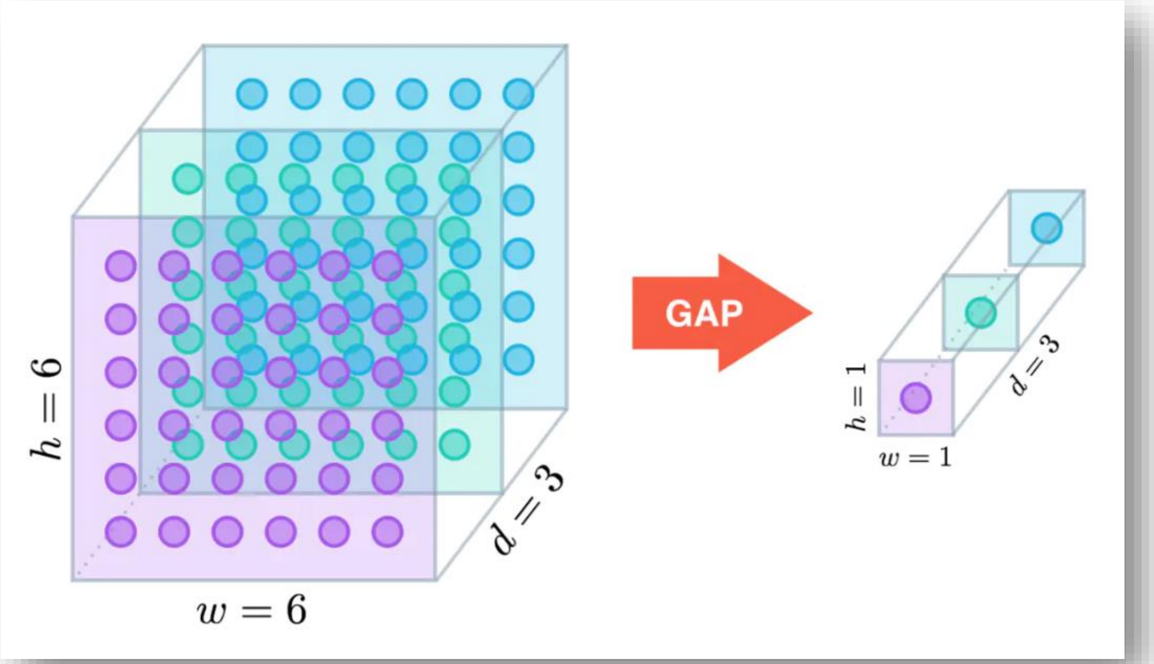
大多数现有的PLL方法，通过利用学习到的模型的内在表征来确定训练过程中的真实标记，依赖于对所收集数据的一些假设。

Background : GAP **全局平均池化**

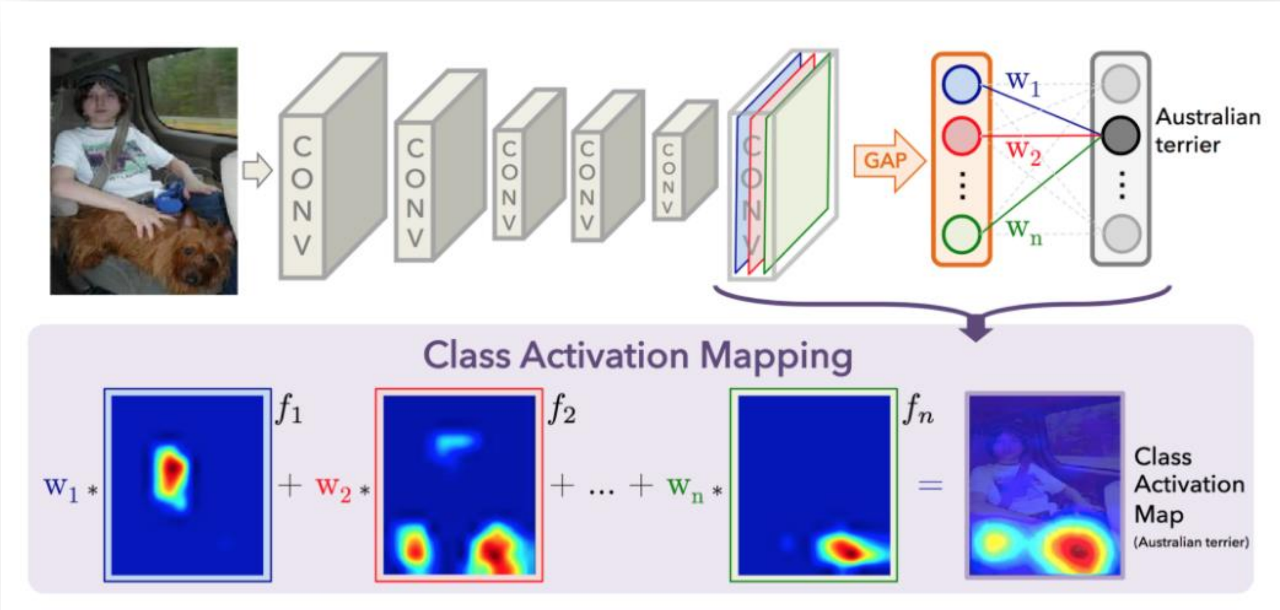
- Global Average Pooling (GAP)

可以看出其是将矩阵中的 [w, h] , 进行平均池化变为一个单元。

相比于**全连接层**有2个优点:
1.模型更轻量化
2.获得目标定位的能力



Background : CAM : Class Activation Map



数学上表示为加权特征图（feature maps）和线性权重（linear weights）的线性组合;

input image： $\boldsymbol{x} \in \mathbb{R}^{d=c\times h\times w}$

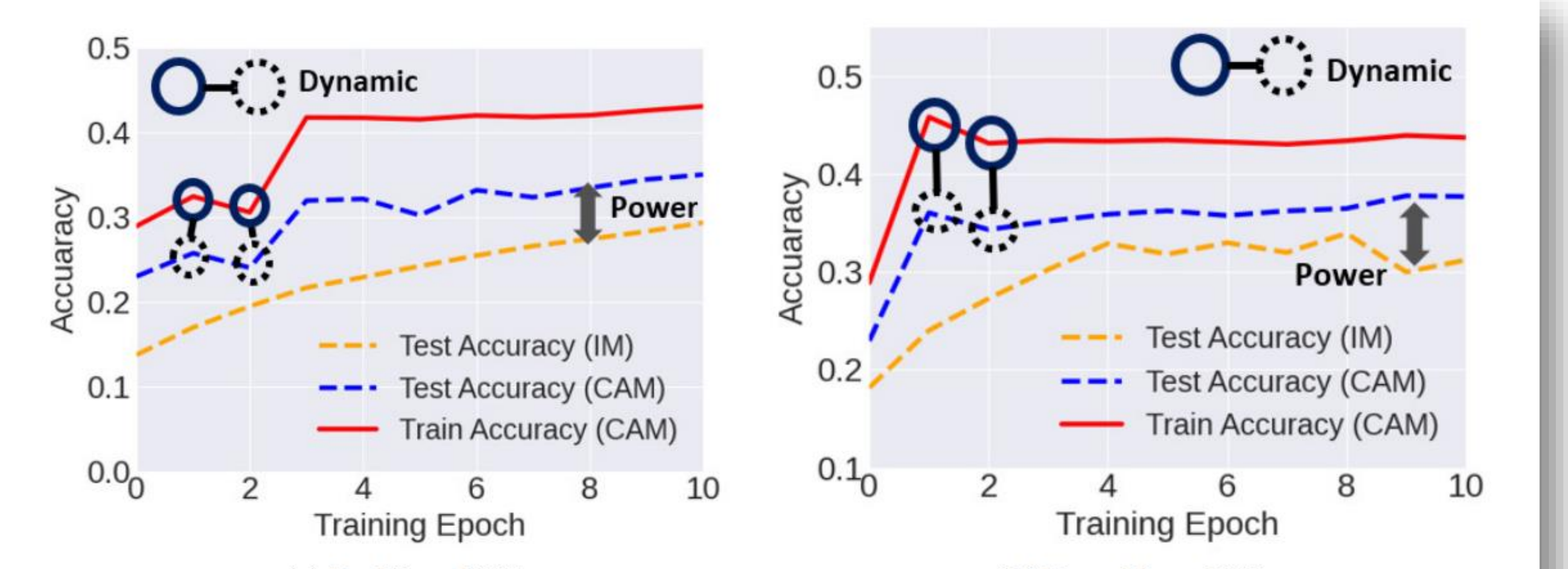
c channels, height ***h***, width ***w***

CAM of x： $\boldsymbol{m} \in \mathbb{R}^{k\times h\times w}$ ：***k*** classes number

$$\boldsymbol{m}^j = \sum_{i=1}^{C_f} (\boldsymbol{\theta}^j)_i e^i(\boldsymbol{x}), j \in \{1, \ldots, k\},$$

Pilot experiment : CAM : Class Activation Maps

Grad-CAM 或 CAM 被视为 f 的内部表示，这样的机制可以引导 f 从候选集中区分出真正的标记。
为了验证我们的猜想，我们通过采用两种不同的标签选择方法进行了试点实验。
1.**直觉法**（IM : intuitive method）； 2. **CAM label 选择策略**



(a) ResNet + USS

(b) DenseNet + FPS

i. CAM可以潜在地引导分类器从候选集中找到真正的标签 -> **power** 属性
ii. CAM识别真实标签的准确性与分类器的准确性同步变化 -> **dynamic** 属性

Grad-CAM : Gradient-weight CAM

为了提升任意的基于CNN的模型 f，提出Grad-CAM 通过合理地利用特定类别梯度信息来实现这种内部表示

x 相对于第 j 个对象的Grad-CAM：$\boldsymbol{g}^j \in \mathbb{R}^{d=c\times h\times w}$

$$\boldsymbol{g}^j = \frac{1}{h \times w} \sum_{z=1}^{c_f} \sum_{m=1}^{h} \sum_{n=1}^{w} \frac{\partial f^j(\boldsymbol{x})}{\partial e^z(\boldsymbol{x})_{m,n}} e^z(\boldsymbol{x}). \tag{2}$$

在本文中，旨在从CAM中提取并改进有用的知识，以解决PLL问题，引导分类器识别真实标记。

CAV : Class Activation Value

动机：为了克服CAM需要CNN这个缺陷；
目的：指导 f 从候选标记集合中识别真正的标记；

受公式（2）中 **Grad-CAM** 的表述启发，将流向最后一个卷积层的梯度信息视为每个神经元对第 j 类的重要性，我们可以直观地推导出每个候选标签的重要性如下。

$$v^j = \left| \frac{\partial(-\log(\psi^j(f(\boldsymbol{x}))))}{\partial f^j(\boldsymbol{x})} \right| \psi^j(f(\boldsymbol{x})) = \left|\psi^j(f(\boldsymbol{x})) - 1\right| \psi^j(f(\boldsymbol{x})), \qquad (3)$$

因为上面的式子有一个缺点：$\psi^j(f(\boldsymbol{x}))$ 相比 $f^j(\boldsymbol{x})$ 来说包含更少的信息，所以换成如下式子

$$v^j = |f^j(\boldsymbol{x}) - 1| f^j(\boldsymbol{x}). \qquad (4)$$

CAVL : Class Activation Value Learning

Partial-Label Learning With Class Activation Value
怎么把CAV用在解决PLL问题上
训练的时候在第一个epoch里有一个小技巧:

$$\hat{\mathcal{R}}_1(\mathcal{L}, f) = \frac{1}{m}\sum\nolimits_{i=1}^{m}\sum\nolimits_{j\in S_i}\mathcal{L}(f(\boldsymbol{x}_i), s_j). \tag{6}$$

为了使分类器不受到来自候选集的假阳性标记的负面影响，由CAV选出潜在的真实标记:

$$y_i' = \arg\max_{j\in S_i} v_i^j, \tag{7}$$

$$\hat{\mathcal{R}}_{\mathrm{cavl}}(\mathcal{L}, f) = \frac{1}{m}\sum\nolimits_{i=1}^{m}\mathcal{L}(f(\boldsymbol{x}_i), y_i'). \tag{8}$$

CAVL : Class Activation Value Learning

Partial-Label Learning With Class Activation Value

Algorithm 1 CAVL Algorithm

Input: Model f, epoch $T_{\max}$, iteration $Z_{\max}$, partial labeled training set $\mathbb{D} = \{(\boldsymbol{x}_i, S_i)\}_{i=1}^{m}$.

```
for t = 1, 2, ..., T_max do
    Shuffle D;
    for z = 1, 2, ..., I_max do
        Sample mini-batch D_z = {(x_j, S_j)}_{j=1}^u from D;
        if t = 1 then
            Update f by minimizing the cross entropy loss function R̂_1 in Eq. (6) with treating all the candidate labels from S_j equally;
        else
            Obtain the CAV v for each x_j by Eq. (4);
            Generate y'_j for x_j by selecting the maximum v from S_j by Eq. (7);
            Update f by minimizing the cross entropy loss function R̂_cavl in Eq. (8) with treating y'_j as the sole true label for each x_j;
        end if
    end for
end for
```

Output: f.

Experiments

Table 1: Test performance of CAVL and other methods on benchmark datasets using data generation by USS. The best results among all methods with the same backbone are marked in **bold**.

Dataset (Backbones)	Method	Accuracy
MNIST (MLP/LeNet)	CC	97.77 ± 0.12% / 98.77± 0.06%
	RC	98.05 ± 0.15% / 99.03 ± 0.04%
	LW	97.99 ± 0.06% / 98.74 ± 0.05%
	PRODEN	97.02 ± 0.08% / 98.81 ± 0.04%
	CAVL	**98.06 ± 0.05% / 99.14 ± 0.03%**
Kuzushiji-MNIST (MLP/LeNet)	CC	88.87 ± 0.32% / 93.83 ± 0.20%
	RC	**89.36 ± 0.30% / 94.01 ± 0.15%**
	LW	87.98 ± 0.39% / 91.52 ± 0.65%
	PRODEN	88.75 ± 0.29% / 93.94 ± 0.18%
	CAVL	88.45 ± 0.22% / 93.25 ± 0.21%
Fashion-MNIST (MLP/LeNet)	CC	87.90 ± 0.27% / 88.96 ± 0.14%
	RC	88.40 ± 0.13% / 89.51 ± 0.11%
	LW	88.16 ± 0.12% / 88.28 ± 0.33%
	PRODEN	88.82 ± 0.15% / 89.23 ± 0.12%
	CAVL	**88.93 ± 0.16% / 89.99 ± 0.10%**
CIFAR-10 (ResNet/DenseNet)	CC	75.74 ± 0.19% / 76.78 ± 0.33%
	RC	77.98 ± 0.46% / 78.56 ± 0.37%
	LW	76.82 ± 0.21% / 78.08 ± 0.66%
	PRODEN	77.62 ± 0.34% / 78.72 ± 0.48%
	CAVL	**78.05 ± 0.32% / 79.10 ± 0.25%**

模型： MLP, LeNet, ReNet, DenseNet

方法： CC, RC, LW, PRODEN, CAVL

Experiments

Table 2: Test performance of CAVL and other methods on benchmark datasets using data generation by FPS. The best results among all methods with the same backbone are marked in **bold**.

Dataset (Backbones)	Method	q=0.3	q=0.5	q=0.7
MNIST (LeNet)	CC	98.87 ± 0.15%	98.49± 0.07%	98.17 ± 0.12%
	RC	98.88 ± 0.07%	98.53± 0.11%	98.12 ± 0.05%
	LW	98.53 ± 0.12%	98.68± 0.06%	97.35 ± 0.13%
	PRODEN	98.72 ± 0.13%	98.62± 0.09%	98.08 ± 0.04%
	CAVL	**98.90 ± 0.12%**	**98.71± 0.04%**	**98.20 ± 0.11%**
Kuzushiji-MNIST (LeNet)	CC	93.11 ± 0.08%	90.87 ± 0.06%	89.98 ± 0.14%
	RC	93.21 ± 0.17%	91.19± 0.22%	90.15. ± 0.04%
	LW	92.65 ± 0.18%	91.28± 0.16%	**90.55 ± 0.17%**
	PRODEN	93.51 ± 0.20%	91.23± 0.17%	90.07 ± 0.08%
	CAVL	**93.82 ± 0.21%**	**91.57± 0.11%**	86.05± 0.25%
Fashion-MNIST (LeNet)	CC	89.41 ± 0.17%	88.72 ± 0.06%	85.87 ± 0.25%
	RC	89.53 ± 0.07%	88.84± 0.14%	85.41 ± 0.18%
	LW	89.19 ± 0.08%	87.19± 0.23%	85.92 ± 0.13%
	PRODEN	89.63 ± 0.05%	88.78± 0.15%	85.81 ± 0.16%
	CAVL	**89.77 ± 0.04%**	**88.92± 0.11%**	**86.25 ± 0.18%**
CIFAR-10 (DenseNet)	CC	77.32 ± 0.14%	76.42 ± 0.13%	66.17 ± 0.25%
	RC	78.14 ± 0.12%	77.42 ± 0.16%	70.21 ± 0.15%
	LW	80.95 ± 0.17%	78.72 ± 0.17%	**71.26 ± 0.16%**
	PRODEN	79.05 ± 0.11%	77.52± 0.18%	70.35 ± 0.18%
	CAVL	**81.58 ± 0.22%**	**79.69 ± 0.17%**	65.86 ± 0.21%

其中q是翻转概率

模型：LeNet, DenseNet

方法：CC, RC, LW, PRODEN

重复 5 次实验，记录了**平均准确率**和**标准偏差**

Experiments

Table 3: Test performance of CAVL and other methods uisng linear model on real-world datasets. The best and second best results among all methods are marked in **bold** and <u>underline</u>.

Method	Real-world datasets				
	Lost	MSRCv2	Birdsong	SoccerPlayer	YahooNews
IPAL	71.16 ± 2.56%	**50.64 ± 3.85%**	70.32 ± 4.85%	55.42 ± 0.92%	66.43 ± 1.32%
CLPL	75.01 ± 4.39%	36.72 ± 4.61%	64.35 ± 1.28%	37.01 ± 1.02%	45.21 ± 0.82%
PLSVM	48.91± 3.33%	35.95 ± 3.96%	48.99 ± 1.98%	45.90 ± 0.98%	57.02 ± 1.02%
PLKNN	37.73± 2.85%	41.28 ± 2.25%	64.32 ± 1.05%	48.21 ± 1.21%	40.67 ± 1.58%
PLECOC	50.95 ± 7.81%	43.25 ± 3.61%	70.51 ± 4.31%	55.29 ± 1.95%	61.23 ± 1.52%
SURE	75.56 ± 5.62%	46.72 ± 4.21%	55.42 ± 1.52%	48.61 ± 0.84%	55.17 ± 1.05%
LW	79.74 ± 3.81%	45.65 ± 3.12%	72.01 ± 2.31%	<u>57.12 ± 3.25%</u>	67.94 ± 1.64%
RC	<u>80.45 ± 3.85%</u>	46.85 ± 2.52%	**72.15 ± 2.14%**	57.05 ± 2.15%	<u>68.12 ± 0.67%</u>
CC	73.78 ± 4.34%	45.13 ± 2.67%	71.86 ± 1.34%	56.54 ± 0.74%	67.00± 0.35%
PRODEN	80.12± 4.52%	45.32 ± 2.38%	71.90 ± 2.34%	56.12 ± 3.12%	67.92± 0.48%
CAVL	**82.07** ± **3.18%**	<u>48.39 ± 2.12%</u>	<u>71.92 ± 0.73%</u>	**57.32** ± **2.24%**	**68.89**± **0.55%**

CAM无法在**BirdSong**这样缺乏空间信息的词嵌入输入中被提取出来。

Experiments

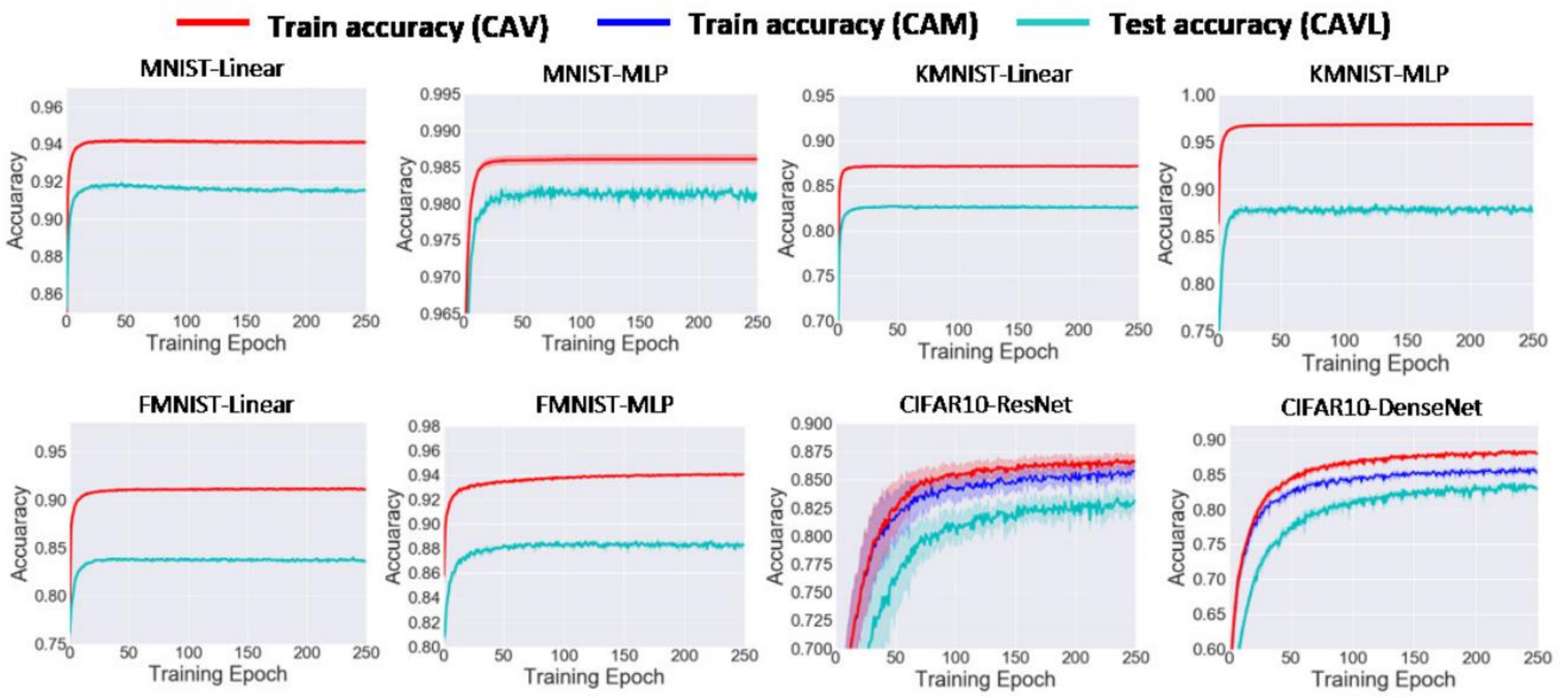


Figure 2: Study of CAV attributes on various backbones and datasets. Note that here CAM could only been obtained in deep CNN architectures, namely ResNet and DenseNet.

Experiments

Table 7: The performance of our CAVL with different settings of the starting epoch.

Starting Epoch	Dataset (Backbones)			
	MNIST (LeNet)	KMNIST (LeNet)	FMNIST (LeNet)	CIFAR-10 (ResNet)
1	**99.14 ± 0.03%**	**93.25 ± 0.21%**	**89.99 ± 0.10%**	**78.05 ± 0.32%**
10	97.82 ± 0.04%	92.84 ± 0.21%	88.75 ± 0.14%	75.27 ± 0.42%
50	97.25 ± 0.03%	91.23 ± 0.25%	87.98 ± 0.15%	72.17 ± 0.31%
100	96.44 ± 0.07%	89.66 ± 0.42%	86.92 ± 0.16%	71.55 ± 0.18%
150	96.38 ± 0.07%	88.54 ± 0.26%	86.07 ± 0.34%	69.87 ± 0.12%

起始epoch之前用IM，后面使用CAV训练；

总结：CAVL的有效性会随着起始历时的延长而减弱。

THANKS