Is normalization indispensable for training deep neural networks?

for medical seminar

Lei Liu

leiliu@link.cuhk.edu.cn

School of Science and Engineering

The Chinese University of Hong Kong, Shenzhen

Aug, 2021

目录

- 简单的回顾.
- 背景和动机.
- 提出的方法.
- 实验.

简单的回顾-Why BN?

例子：线性回归 (房价预测问题)

Living Area (feet^2)	Price (1000$)
5719	567
3241	345
1139	141
1572	167
1101	287
2576	227
⋮	⋮

图: 数据

例: 线性回归

- 输入数据: $x_1, \cdots, x_n$, where $n = 17$
- 回归模型: w_0, w_1, $p = 2$ (two dimension)
- 输入矩阵: $X \in \mathbb{R}^{p \times n}$, 第一行是数据 (Area), 第二行全 1
- 标准输出: $y \in \mathbb{R}^{n \times 1}$, 房屋价格 (price)
- 优化问题: $min \frac{1}{2} \|X^T w - y\|^2$

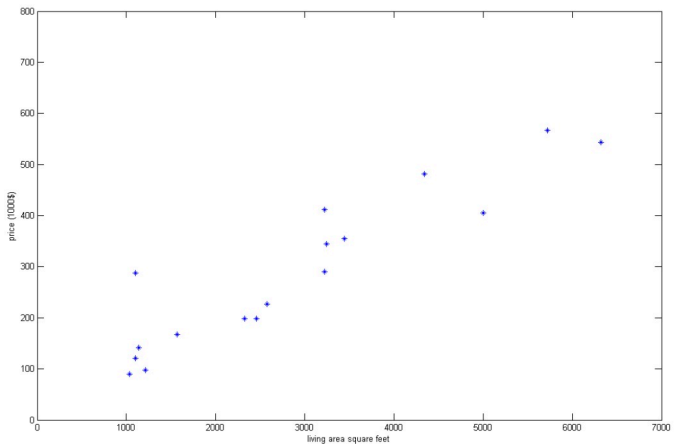

图: 问题定义

例: 问题求解

- rescale 数据: Area*0.01, Price*0.1
- 1000 次梯度下降迭代; 初始参数值 (10, 50)

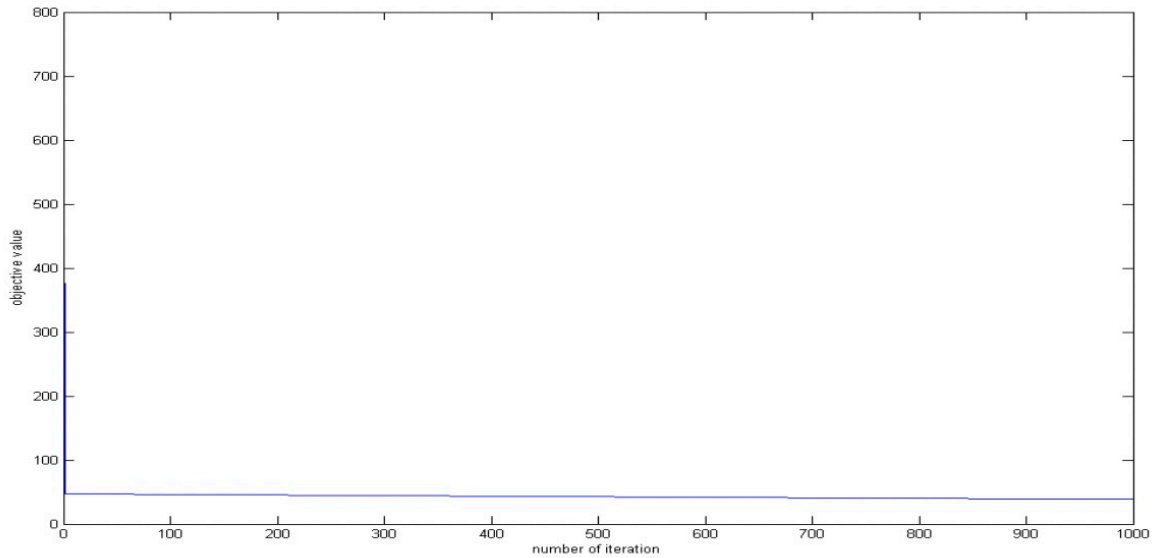

图: 目标函数值

例: 问题求解

- 结果: 远离最优解
- 分析: 没有收敛

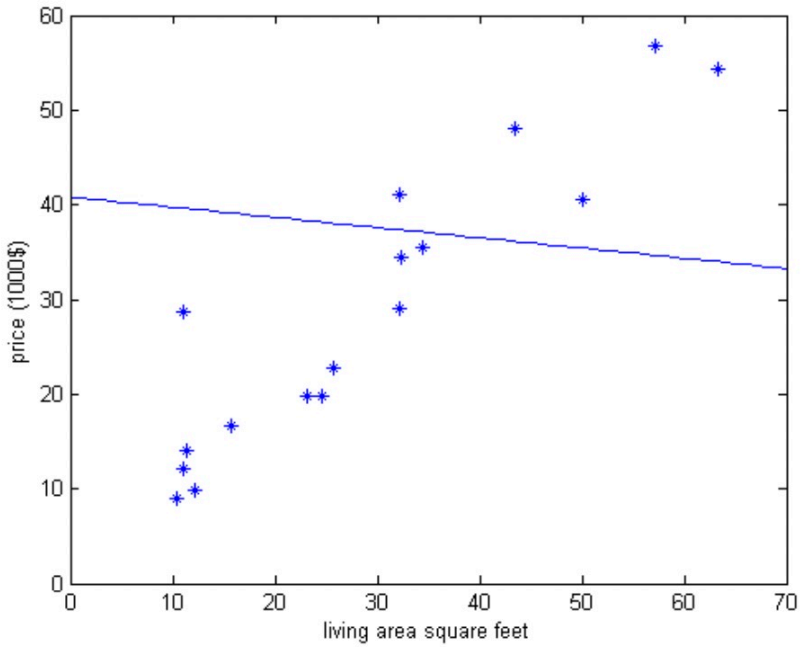

图: 拟合函数

Why ? XX^T 的特征值分别是 0.004 和 1.856. ill-conditioned Hessian!

例: 问题求解

- rescale 数据: Area*0.0001, Price*0.1
- 1000 次梯度下降迭代; 初始参数值 (10, 50)

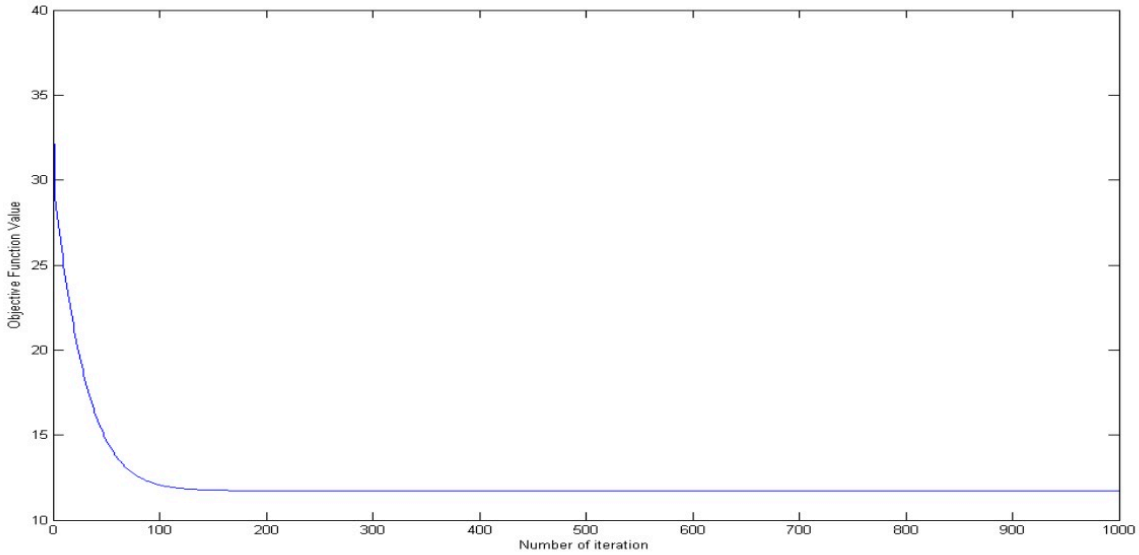

图: 目标函数值

Why does it work now?
XX^T 的特征值分别是 0.4 和 18.45. well-conditioned Hessian!

例: 问题求解

- 结果: 接近最优解
- 分析: 收敛

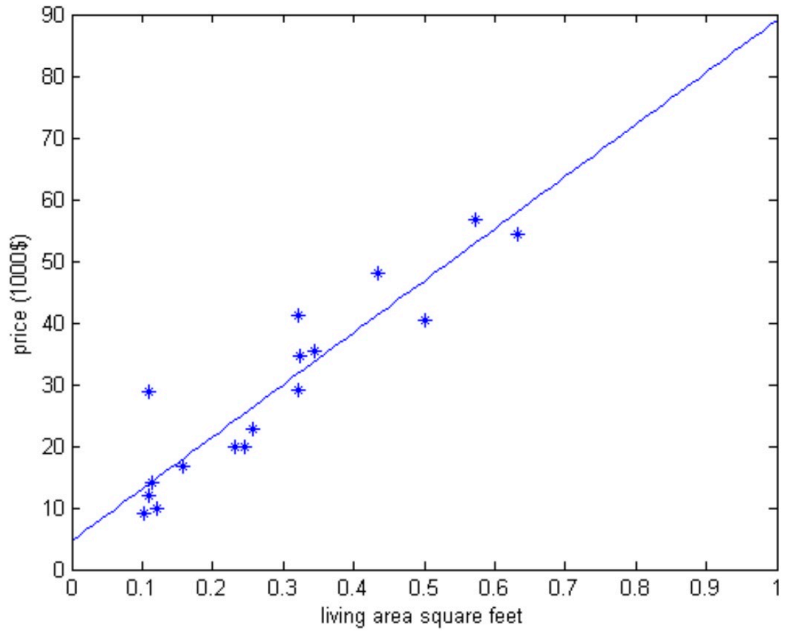

图: 拟合函数

Why ? 数据标准化可以减小 condition number, 加快收敛速度

BN 定义

Batch Normalization 的定义如下:

$$\mathrm{BN}_{\gamma,\beta}(a_1,\ldots,a_N) \triangleq \left(\gamma\frac{a_1-\mu}{\sigma+\epsilon}+\beta,\ldots,\gamma\frac{a_N-\mu}{\sigma+\epsilon}+\beta\right),$$

$$\text{where } \mu=\frac{1}{N}(a_1+\cdots+a_N), \quad \sigma=\sqrt{\frac{1}{N}\sum_{i=1}^{N}(a_i-\mu)^2}.$$

图: BN

γ, β 是可学习的参数 (为了保证 representation power 不减弱)

其他优点

- 缓解梯度弥散和梯度爆炸 [Ioffe and Szegedy,2015]
- 分析: 减小模型激活值分布的协方差偏移

其他变种

- Layer Normalization for recurrent neural networks [Ba, Kiros, and Hinton, 2016]
- Instance Normalization for stylization [Ulyanov, Vedaldi, and Lempitsky, 2016]
- Group Normalization (GN) for small-batch training [Wu and He, 2018]

背景和动机

BN 仍然被当作一个黑盒来研究，因此

- 如何不使用 BN 进行稳定的训练？
- 如何不使用 BN 进行高准确率的训练？

第一个问题:

- 稳定的训练: Fix-up initialization
- 但存在较小的性能下降

本文要研究的问题是第二个问题

Preliminaries and Notations

模型结构: ResNet 残差连接计算如下:

$$x_\ell = x_{\ell-1} + F_\ell(x_{\ell-1}) \tag{1}$$

- x_0: 输入
- x_l: 第 ℓ 个残差块 (F_l) 的输出
- x_l: Activation (默认 ReLUs)

存在的问题 (方差爆炸)

- 随机初始化: 随着网络深度加深，训练不稳定，信号的方差变大 (梯度消失、爆炸)
- Kaiming 初始化: 输入的方差和残差块输出的方差一致 $Var(F_\ell(x_{\ell-1})) = Var(x_{\ell-1})$

$$
\begin{aligned}
\operatorname{Var}\left(\boldsymbol{x}_{\ell}\right) &= \operatorname{Var}\left(\boldsymbol{x}_{\ell-1}+\mathcal{F}_{\ell}\left(\boldsymbol{x}_{\ell-1}\right)\right) \\
&\approx \operatorname{Var}\left(\boldsymbol{x}_{\ell-1}\right)+\operatorname{Var}\left(\mathcal{F}_{\ell}\left(\boldsymbol{x}_{\ell-1}\right)\right) \\
&= 2 \operatorname{Var}\left(\boldsymbol{x}_{\ell-1}\right)
\end{aligned}
\tag{2}
$$

存在的问题 (Dead Relu)

描述: 某些神经元在整个训练过程中没有被激活
线性层: $y_k = W_k x_k + b_k$, $x_k = ReLU(y_{k-1})$, 使用 Kaiming 初始化来初始化 W_k, $w_{ij}^k \sim \mathcal{N}(0, 2/d)$, $b_k = 0$. 假设 x_k 元素的期望为 c_k.

$$\mathbb{E}\left([\boldsymbol{y}_k]_i\right) = \mathbb{E}\left(\sum_{j=1}^{d} w_{ij}^k \left[\boldsymbol{x}_k\right]_j\right) = \mathrm{c}_k W_i^k, \text{ where } W_i^k = \sum_{j=1}^{d} w_{ij} \quad (3)$$

- c_k 非负,
- $W_i^k \sim \mathcal{N}(0, 2)$, 为负的概率很高
- 仿真实验中，40% 的神经元没有被激活 (20 layers)

提出的方法 (Re-scaling)

$$\boldsymbol{x}_k = \alpha_k \boldsymbol{x}_{k-1} + \beta_k \mathcal{F}_k\left(\boldsymbol{x}_{k-1}\right) \tag{4}$$

where $\alpha_2^k + \beta_2^k = 1$.

- $\mathcal{F}_k$ 未经过 normalization, 使用 kaiming 初始化
- $\mathrm{Var}\left[\mathcal{F}_k\left(\boldsymbol{x}_{k-1}\right)\right] \approx \mathrm{Var}\left[\boldsymbol{x}_{k-1}\right]$
- $\mathrm{Var}\left[\boldsymbol{x}_k\right] = \alpha_k^2 \mathrm{Var}\left[\boldsymbol{x}_{k-1}\right] + \beta_k^2 \mathrm{Var}\left[\mathcal{F}_k\left(\boldsymbol{x}_{k-1}\right)\right] = \mathrm{Var}\left[\boldsymbol{x}_{k-1}\right]$

考虑最终的输出 x_L

$$\boldsymbol{x}_L = \left(\prod_{i=1}^{L} \alpha_i\right) \boldsymbol{x}_0 + \sum_{k=1}^{L} \beta_k \prod_{i=k+1}^{L} \alpha_i \mathcal{F}_k\left(\boldsymbol{x}_{k-1}\right) \tag{5}$$

提出的方法 (Re-scaling)

最优的相关系数需要保证不同残差块的参数一致

$$\forall k \neq k', \beta_k \prod_{i=k+1}^{L} \alpha_i = \beta_{k'} \prod_{i=k'+1}^{L} \alpha_i \tag{6}$$

- $\alpha_k = \sqrt{(k-1+c)/(k+c)}$, $\beta_k = 1/\sqrt{k+c}$, c 为超参数

公式 (6) 可以避免不稳定的方差。但是对于训练过程，仍然要控制梯度的大小防止梯度剧烈变化。

$$\Delta\ell = \ell\left(\boldsymbol{w}_{t+1}\right) - \ell\left(\boldsymbol{w}_t\right) = \ell\left(\boldsymbol{w}_t - \eta\nabla\ell\left(\boldsymbol{w}_t\right)\right) - \ell(\boldsymbol{w}) = -\eta\left\|\nabla\ell\left(\boldsymbol{w}_t\right)\right\|_2^2 + O\left(\eta^2\right) \tag{7}$$

如果梯度项值很大，需要使用较小的学习率。否则容易造成输出溢出。

提出的方法 (Re-scaling)

$$
\begin{aligned}
\|\nabla \ell\|_2^2 &= \sum_{k=1}^{L} \left(\frac{\partial \ell}{\partial \boldsymbol{x}_k} \cdot \sqrt{\frac{1}{k+c}} \frac{\partial \mathcal{F}_k}{\partial \Theta_k} \right)^2 \\
&\approx \sum_{k=1}^{L} \frac{1}{k+c} \left\| \frac{\partial \ell}{\partial \boldsymbol{x}_k} \right\|_2^2 \left\| \frac{\partial \mathcal{F}_k}{\partial \Theta_k} \right\|_F^2 \\
&= O\left(\sum_{k=1}^{L} \frac{1}{k+c} \right)
\end{aligned}
\tag{8}
$$

- 如果 c 很小, 梯度的大小和网络深度有关。($\|\nabla \ell\|_2^2 = O(InL)$)
- 如果 $c = L$ 很小, 梯度的大小和网络深度的关系减弱。

$$
\boldsymbol{x}_k = \sqrt{\frac{k-1+L}{k+L}} \boldsymbol{x}_{k-1} + \sqrt{\frac{1}{k+L}} \mathcal{F}_k \left(\boldsymbol{x}_{k-1}, \Theta_k \right) \tag{9}
$$

提出的方法 (Scalar multipliers)

c 的取值会影响到不同层网络参数的学习

- 如果 c 的取值小, 深层网络的权重较大
- 如果 c 的取值大, 浅层网络的权重较大

Motivation: 随着训练过程, 网络模型的学习过程是从浅到深的。即学习重心是从浅层网络转移到深层网络的。

$$\boldsymbol{x}_k = \sqrt{\frac{k-1+L}{k+L}}\boldsymbol{x}_{k-1} + \frac{m_k}{\sqrt{L}}\mathcal{F}_k\left(\boldsymbol{x}_{k-1}\right) \tag{10}$$

实验

Class-Center Involved Triplet Loss for Skin Disease Classification on Imbalanced Data

Method	Regularization	Accuracy
Fixup-Init	None [40]	72.4%
	Mixup [40]	76.0%
	Dropout	75.5%
SkipInit	None [7]	74.9%
	Dropout [7]	75.6%
RescaleNet	None	74.3%
	Mixup	76.4%
	Dropout	**76.6%**

Table 2: ResNet50 validation accuracies on ImageNet for non-normalization methods.

Method	Accuracy(%)
Batch Normalization [17]	76.4
Layer Normalization [2]	74.7
Instance Normalization [33]	71.6
Group Normalization [36]	75.9
Switchable Normalization [21]	76.9
RescaleNet	76.58±0.08
Filter Response Norm [31]	77.2
RescaleNet + Cosine LR	**77.2±0.06**

Table 3: ResNet50 Validation accuracies on ImageNet compared with normalization methods.

图: 实验结果

Reference I


Sergey Ioffe and Christian Szegedy. “Batch normalization: Accelerating deep network training by reducing internal covariate shift”. In: **International conference on machine learning**. PMLR. 2015, pp. 448–456.


Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. “Layer normalization”. In: **arXiv preprint arXiv:1607.06450** (2016).


Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. “Instance normalization: The missing ingredient for fast stylization”. In: **arXiv preprint arXiv:1607.08022** (2016).


Yuxin Wu and Kaiming He. “Group normalization”. In: **Proceedings of the European conference on computer vision (ECCV)**. 2018, pp. 3–19.

Class-Center Involved Triplet Loss for Skin Disease Classification on Imbalanced Data

for medical seminar

Lei Liu

leiliu@link.cuhk.edu.cn

School of Science and Engineering

The Chinese University of Hong Kong, Shenzhen

Aug, 2021

目录

背景和动机

- 数据不平衡是医疗图像诊断领域中常见的问题
- 常用的方法为重加权和重采样
- 本文从度量学习的角度进行分析

提出的方法

$$\|\mathrm{f}(\mathrm{x}_i;\boldsymbol{\theta})-\mathrm{f}(\mathrm{y}_i;\boldsymbol{\theta})\|+\alpha<\|\mathrm{f}(\mathrm{x}_i;\boldsymbol{\theta})-\mathrm{f}(\mathrm{z}_i;\boldsymbol{\theta})\| \tag{1}$$

- x_i, y_i 为正样本对
- x_i, z_i 为负样本对
- θ 是模型参数
- $\mathrm{f}(\cdot;\boldsymbol{\theta})$ 是模型提取的特征
- $\|\cdot\|$ 范数
- α 是正样本对和负样本对之间的距离间隔

$$\begin{aligned} l(\mathrm{x}_i,\mathrm{y}_i,\mathrm{z}_i;\boldsymbol{\theta}) = [&\|\mathrm{f}(\mathrm{x}_i;\boldsymbol{\theta})-\mathrm{f}(\mathrm{y}_i;\boldsymbol{\theta})\|+\alpha \\ &-\|\mathrm{f}(\mathrm{x}_i;\boldsymbol{\theta})-\mathrm{f}(\mathrm{z}_i;\boldsymbol{\theta})\|]_+ \end{aligned} \tag{2}$$

提出的方法

$$\|\mathrm{f}(\mathrm{x}_i;\boldsymbol{\theta})-\mathrm{f}(\mathrm{y}_i;\boldsymbol{\theta})\|+\alpha<\|\mathrm{f}(\mathrm{x}_i;\boldsymbol{\theta})-\mathrm{f}(\mathrm{z}_i;\boldsymbol{\theta})\| \tag{3}$$

$$\|\mathrm{f}(\mathrm{x}_i;\boldsymbol{\theta})-\mathrm{f}(\mathrm{y}_j;\boldsymbol{\theta})\|+\alpha<\|\mathrm{f}(\mathrm{x}_i;\boldsymbol{\theta})-\mathrm{f}(\mathrm{z}_j;\boldsymbol{\theta})\| \tag{4}$$

在大类别中, 特征的分布偏差较大, y_i, y_j(huozh) 的巨大差异可以会导致训练不稳定
使用全局信息 (样本特征中心) 来进行度量学习

$$\begin{aligned} l(\mathrm{x}_i,\mathrm{y}_i,\mathrm{z}_i;\boldsymbol{\theta}_t)=[&\|\mathrm{f}(\mathrm{x}_i;\boldsymbol{\theta}_t)-\mathrm{c}(\mathrm{x}_i;\boldsymbol{\theta}_{t-1})\|+\alpha \\ &-\|\mathrm{f}(\mathrm{x}_i;\boldsymbol{\theta}_t)-\mathrm{c}(\mathrm{z}_i;\boldsymbol{\theta}_{t-1})\|]_+ \end{aligned} \tag{5}$$

实验

Table 2. Comparisons between the proposed approach with baseline methods on Skin7 and Skin198 datasets.

	Skin7						Skin198					
	BCE	WCE	OCE	WFCE	TP (Ours)	TPC (Ours)	BCE	WCE	OCE	WFCE	TP (Ours)	TPC (Ours)
MF1	83.65 (1.52)	82.45 (1.31)	83.53 (1.33)	83.52 (1.63)	84.31 (1.93)	**84.89** (0.91)	51.91 (1.10)	60.21 (1.36)	59.77 (1.89)	53.28 (2.65)	61.90 (1.80)	**63.21** (1.61)
Precision	86.96 (1.96)	83.35 (1.79)	87.26 (1.27)	86.43 (1.34)	88.31 (1.79)	**88.42** (0.62)	56.41 (1.27)	64.82 (1.34)	64.87 (2.06)	58.31 (2.77)	**66.11** (2.03)	65.55 (1.65)
Recall	81.15 (1.62)	82.06 (1.47)	80.81 (1.39)	81.25 (1.78)	81.11 (2.27)	**83.02** (0.71)	52.12 (1.14)	60.23 (1.12)	59.34 (1.87)	53.34 (2.58)	62.10 (1.81)	**64.68** (1.63)

Table 3. Performance of methods on small-samples classes of Skin7 and Skin198.

	Skin7						Skin198					
	BCE	WCE	OCE	WFCE	TP (Ours)	TPC (Ours)	BCE	WCE	OCE	WFCE	TP (Ours)	TPC (Ours)
MF1	73.67 (3.62)	77.96 (5.31)	74.05 (8.91)	76.21 (4.94)	75.58 (5.60)	**81.22** (5.07)	18.59 (2.43)	53.37 (1.99)	56.41 (3.55)	20.36 (2.08)	59.31 (3.36)	**61.03** (2.84)
Precision	79.03 (0.76)	87.18 (2.47)	84.93 (5.16)	84.96 (3.61)	86.89 (9.09)	**89.45** (6.97)	24.22 (3.00)	65.21 (2.52)	66.46 (4.25)	26.83 (2.74)	**66.68** (4.03)	63.69 (3.24)
Recall	69.39 (6.24)	70.83 (7.64)	66.17 (11.81)	69.35 (6.62)	67.78 (8.03)	**75.30** (8.39)	16.67 (2.78)	49.79 (2.68)	53.42 (3.17)	17.99 (2.21)	58.45 (3.14)	**63.87** (3.00)

图: 实验结果

Thank You !