ICML2021 | Self-Tuning: 如何减少对标记数据的需求？

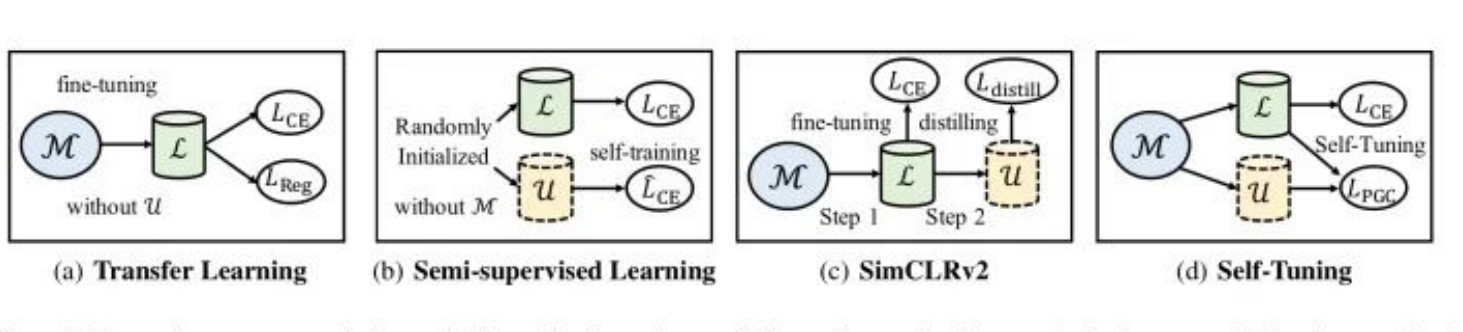
夕小瑶的卖萌屋 2021-07-27 19:01

以下文章来源于THUML，作者王希梅

THUML
清华大学软件学院机器学习实验室，专注于迁移学习、深度学习、知识学习等基础理...

文 | 王希梅，高敬涵，龙明盛，王建民
源 | THUML

本文介绍ICML2021的中稿论文：Self-Tuning for Data-Efficient Deep Learning，就“如何减少对标记数据的需求”这一重要问题给出了我们的思考。



(a) **Transfer Learning** (b) **Semi-supervised Learning** (c) **SimCLRv2** (d) **Self-Tuning**

Figure 1. Comparisons among techniques. (a) **Transfer Learning**: only fine-tuning on $\mathcal{L}$ with a regularization term; (b) **Semi-supervised Learning**: a common practice for SSL is a CE loss on $\mathcal{L}$ while self-training on $\mathcal{U}$ without a decent pretrained model; (c) **SimCLRv2**: fine-tune model $\mathcal{M}$ on $\mathcal{L}$ first and then distill on $\mathcal{U}$; (d) **Self-Tuning**: unify the exploration of $\mathcal{L}$ and $\mathcal{U}$ and the transfer of model $\mathcal{M}$.

论文标题：

Self-Tuning for Data-Efficient Deep Learning

论文链接：

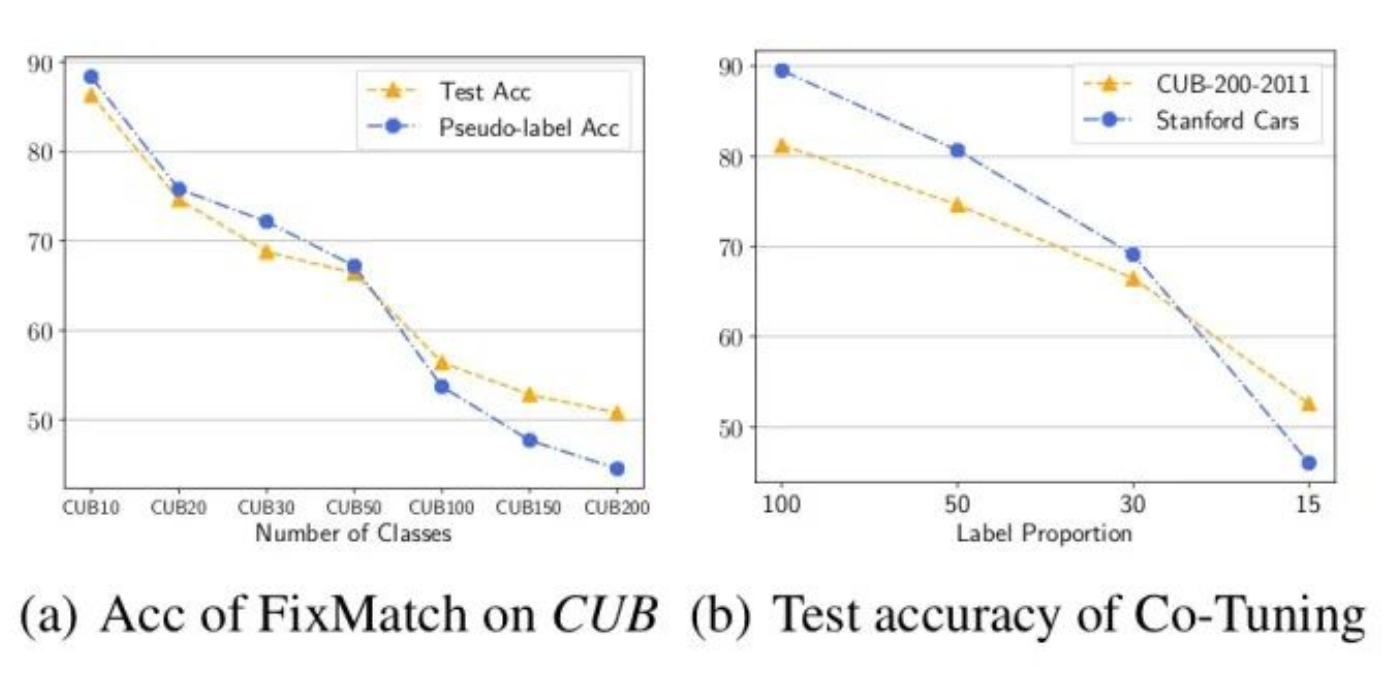
http://ise.thss.tsinghua.edu.cn/~mlong/doc/Self-Tuning-for-Data-Efficient-Deep-Learning-icml21.pdf

GitHub链接：

https://github.com/thuml/Self-Tuning

引言

大规模标记数据集推动深度学习获得了广泛应用，然而，在现实场景中收集足量的标记数据往往耗时耗力。为了减少对标记数据的需求，半监督学习和迁移学习的研究者们从两个不同的视角给出了自己的思考：半监督学习(Semi-supervised Learning, SSL)侧重于同时探索标记数据和无标记数据，通过挖掘无标记数据的内在结构增强模型的泛化能力，而迁移学习(Transfer Learning, TL)旨在将预训练模型微调到目标数据中，也就是我们耳熟能详的**预训练-微调**范式。



(a) Acc of FixMatch on *CUB* (b) Test accuracy of Co-Tuning

Figure 4. Test accuracy of a state-of-the-art SSL method and a TL method on various class numbers or label ratios respectively.

半监督学习的最新进展，例如UDA，FixMatch等方法，证明了自训练(Self-Training)的巨大潜力。通过弱增广样本为强增广样本生成伪标记(pseudo-label)，FixMatch就可以在Cifar10、SVHN、STL-10数据集上取得了令人耳目一新的效果。然而，细心的读者会发现，上述数据集都是类别数较少的简单数据集(都是10类)，当类别数增加到100时，FixMatch这种从头开始训练(train from scratch)的自训练方法的表现就差强人意了。进一步地，我们在CUB200上将类别数从10逐渐增加到200时，发现FixMatch的准确率随着伪标签的准确率的下降而快速下降。这说明，随着类别数的增加，伪标签的质量逐渐下降，而自训练的模型也被错误的伪标签所误导，从而难以在测试数据集上取得可观的效果。这一现象，被前人总结为自训练的**确认偏差**(confirmation bias)问题，说明Self-training虽然是良药，偶尔却有毒。

迁移学习在计算机视觉和自然语言处理中被广泛使用，预训练-微调(fine-tuning)的范式也比传统的领域适应(domain adaptation)约束更少，更具落地价值。然而，现有的迁移学习方法专注于从不同角度挖掘预训练模型和标记数据，却对更为容易获取的无标记数据熟视无睹。以迁移学习的最新方法Co-Tuning为例，它通过学习源领域类别和目标领域类别的映射关系，实现了预训练模型参数的完全迁移。然而，因为仅仅将预训练模型迁移到标记数据中，Co-Tuning容易过拟合到有限的标记数据上，测试准确率随着标记数据比例的减少而迅速下降，我们将这一现象总结为**模型漂移**(model shift)问题。

为了摆脱迁移学习和半监督学习的困境，我们提出了一种称为数据高效深度学习(data-efficient deep learning)的新场景，通过同时挖掘预训练模型和无标记数据的知识，使他们的潜力得到充分释放。这可能是迁移学习在工业界最为现实的落地场景：当我们试图获得目标领域的一个优秀模型时，源领域的预训练模型和目标领域的无标记数据几乎唾手可得。同时，为了解决前述的确认偏差和模型漂移问题，我们提出了一种称为Self-Tuning的新方法，将标记数据和无标记数据的探索与预训练模型的迁移融为一体，以及一种通用的伪标签组对比机制(Pseudo Group Contrast)，从而减轻对伪标签的依赖，提高对伪标签的容忍度。在多个标准数据集的实验表明，Self-Tuning远远优于半监督学习和迁移学习的同类方法。例如，**在标签比例为15%的Stanford-Cars数据集上，Self-Tuning的测试精度比fine-tuning几乎提高了一倍**。

如何解决确认偏差问题?

为了找出自训练的**确认偏差**(confirmation bias)问题的根源，我们首先分析了伪标签(pseudo-label)广泛采用的交叉熵损失函数(**Cross-Entropy, CE**):

$$\widehat{L}_{\text{CE}} = -\sum_{c=1}^{C} \mathbf{1}(\widehat{y}_i = c)\mathbf{1}(z_i > t)\log \mathbf{p}_i^c$$

其中，$\hat{y}_i$是输入$\mathbf{x}_i$生成的伪标签， 而$z_i = \max_c(\mathbf{p}_i^c)$是模型对于样本$\mathbf{x}_i$。通常地，大多数自训练方法都会针对confidence做一个阈值过滤，只有大于阈值t (比如FixMatch中设置了0.95的阈值)的样本的预测标签才会被视为合格的伪标签加入模型训练。然而，如图2所示，由于交叉熵损失函数专注于学习不同类别的分类面，如果某些伪标签存在错误，通过交叉熵损失函数训练的模型就会轻易地被错误的伪标签所误导。

为了解决交叉熵损失函数的**类别鉴别**(class discrimination)特性对自训练带来的挑战，最近取得突破进展的基于**样本鉴别**(sample discrimination)思想的对比学习损失函数吸引了我们的注意。给定由输入$\mathbf{x}_i$生成的查询样本$\mathbf{q}$，$\mathbf{x}_i$在不同数据增广下生成的副本$\mathbf{k}_0$，以及D个不同输入生成的负样本$\{\mathbf{k}_1, \mathbf{k}_2, \cdots, \mathbf{k}_D\}$，则通过内积度量相似性的对比学习(**Constrastive Learning, CL**)损失函数可以定义为

$$L_{\text{CL}} = -\log \frac{\exp(\mathbf{q}\cdot\mathbf{k}_0/\tau)}{\exp(\mathbf{q}\cdot\mathbf{k}_0/\tau) + \sum_{d=1}^{D}\exp(\mathbf{q}\cdot\mathbf{k}_d/\tau)}$$

可以看出，对比学习旨在最大化同一样本在两个不同数据增广下的表征相似性，而最小化不同样本间的表征相似性，从而实现样本鉴别，挖掘数据中隐藏的流形结构。这种设计与伪标签无关，天然地不受错误的伪标签的影响。然而，标准的对比学习损失函数未能将标签和伪标签嵌入到模型训练中，从而使有用的鉴别信息束之高阁。

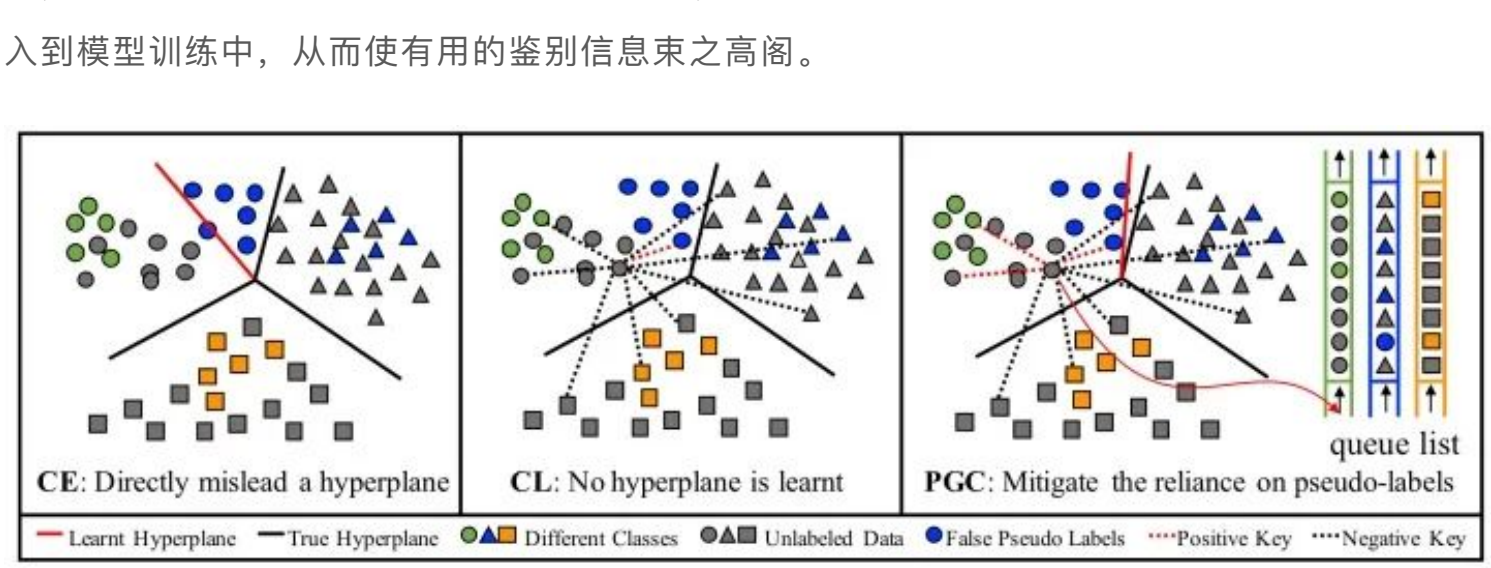


Figure 2. Comparison of various loss functions: (a) **CE**: cross-entropy loss will be easily misled by false pseudo-labels; (b) **CL**: contrastive learning loss underutilizes labels and pseudo-labels; (c) **PGC**: Pseudo Group Contrast mechanism will mitigate the confirmation bias.

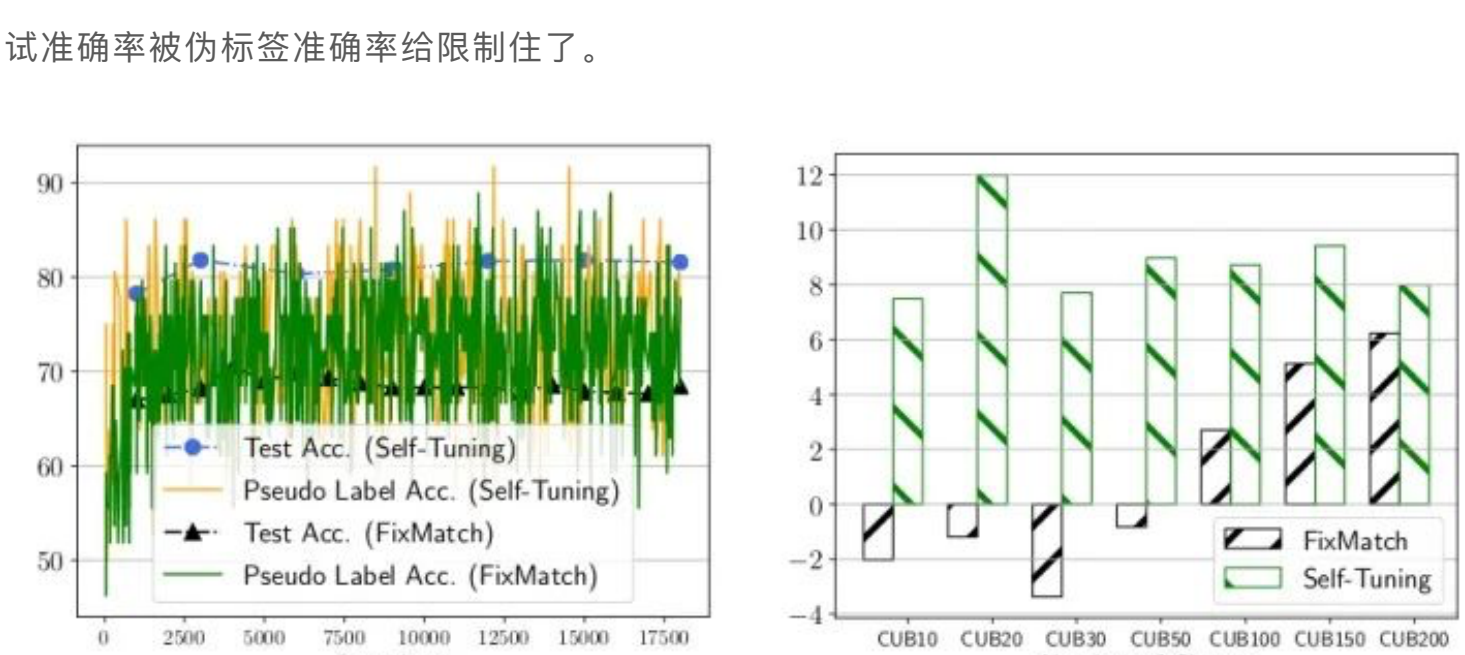
为了解决这一挑战，我们提出了一种通用的伪标签组对比机制(**Pseudo Group Contrast, PGC**)。对于任何一个查询样本$\mathbf{q}$，它的伪标签用$\hat{y}$表示。PGC将具有相同伪标签($\hat{y}$)的样本都视为正样本，而具有不同伪标签($\{1, 2, \cdots, C\}\setminus\hat{y}$)的样本则组成了负样本，从而最大化查询样本与具有相同伪标签的正样本的表征相似性，实现伪标签的组对比。

$$\widehat{L}_{\text{PGC}} = -\frac{1}{D+1}\sum_{d=0}^{D}\log\frac{\exp(\mathbf{q}\cdot\mathbf{k}_d^{\widehat{y}}/\tau)}{\text{Pos}+\text{Neg}}$$

$$\text{Pos} = \exp(\mathbf{q}\cdot\mathbf{k}_0^{\widehat{y}}/\tau) + \sum_{j=1}^{D}\exp(\mathbf{q}\cdot\mathbf{k}_j^{\widehat{y}}/\tau)$$

$$\text{Neg} = \sum_{c=1}^{\{1,2,\cdots,C\}\setminus\widehat{y}}\ \sum_{j=1}^{D}\exp(\mathbf{q}\cdot\mathbf{k}_j^{c}/\tau),$$

那么，为什么PGC机制就可以提高对错误的伪标签的容忍度呢？我们认为，这是因为PGC采用了具有竞争机制的softmax函数，同一伪标签下的正样本会互相竞争。如果正样本的伪标签是错误的，这些伪标签的样本也会在竞争中落败，因为那些具有正确伪标签的正例样本的表征与查询样本的表征更相似。这样的话，模型在训练过程中会更多地受到正确的伪标签的影响，而不是像交叉熵损失函数那样直接地受到错误的伪标签的误导。我们在CUB数据集上的分析实验也证明了这一点：1. 在模型训练伊始，Self-Tuning和FixMatch具有相似的伪标签准确率，但是随着模型逐渐趋于收敛，Self-Tuning的测试集准确率明显高于FixMatch。2. 在具有不同类别数的CUB数据集上，Self-Tuning的测试准确率始终高于伪标签准确率，而FixMatch的测试准确率被伪标签准确率给限制住了。



(a) Training Process on *CUB30* (b) $\text{Acc}_{\text{test}} - \text{Acc}_{\text{pseudo\_labels}}$

如何解决模型漂移问题?

如前所述，当我们只在有限的标记数据集上微调预训练模型时，模型漂移问题往往难以避免。为了解决这个问题，近期发表的一篇名为SimCLRv2的论文提出可以综合利用预训练模型、标记数据和无标记数据的信息。他们给出了一个有趣的解决方案：首先在标记数据集($\mathcal{L}$)上微调预训练模型($\mathcal{M}$)，继而在无标记数据集($\mathcal{U}$)上进行知识蒸馏。然而，通过这一从$\mathcal{M}$到$\mathcal{L}$再到$\mathcal{U}$的“序列化”方式，微调后的模型依然倾向于向有限的标记数据偏移。我们认为，应该将标记和未标记数据的探索与预训练模型的迁移统一起来。

与SimCLRv2的“序列化”方式不同，我们提出了一种“一体化”的形式来解决模型漂移问题。首先，与半监督学习从零开始训练模型的通用实践不同，Self-Tuning的模型起点是一个相对准确的大规模预训练模型，通过更准确的初始化模型来提供一个更好的隐式正则。同时，预训练模型的知识将并行地流入标记数据和无标记数据中，标记数据和无标记数据产生的梯度也会同时更新模型参数。这种“一体化”的形式有利于同时探索标记数据的判别信息和无标记数的内在结构，大大缓解模型漂移的挑战。

另一方面，在对比学习中，负样本的规模越大，模型的效果往往越好。与MoCo类似，我们也通过引入队列的方式将负样本规模与批量大小(batch-size)解耦，使得负样本规模可以远大于批量大小。另一方面，队列的方式可以保证每次对比时，每个伪类下的负样本数目恒定，不受每个minibatch随机采样的影响。与标准的对比学习不同的是，由于伪标签的引入，PGC需要维护C个队列，其中C是类别数。在每次模型迭代中，对于无标记样本，将根据他们的伪标签渐进地替换对应队列里面最早的样本。而对于标记数据，因为他们天然地拥有准确的标签，则可以根据他们的标签来更新对应的队列。值得注意的是，我们在标记数据和无标记数据间共享了这些队列。这一设计的好处在于：将标记数据中宝贵的准确标签嵌入到共享队列中，从而提高了无标记数据的候选样本的伪标签准确性。

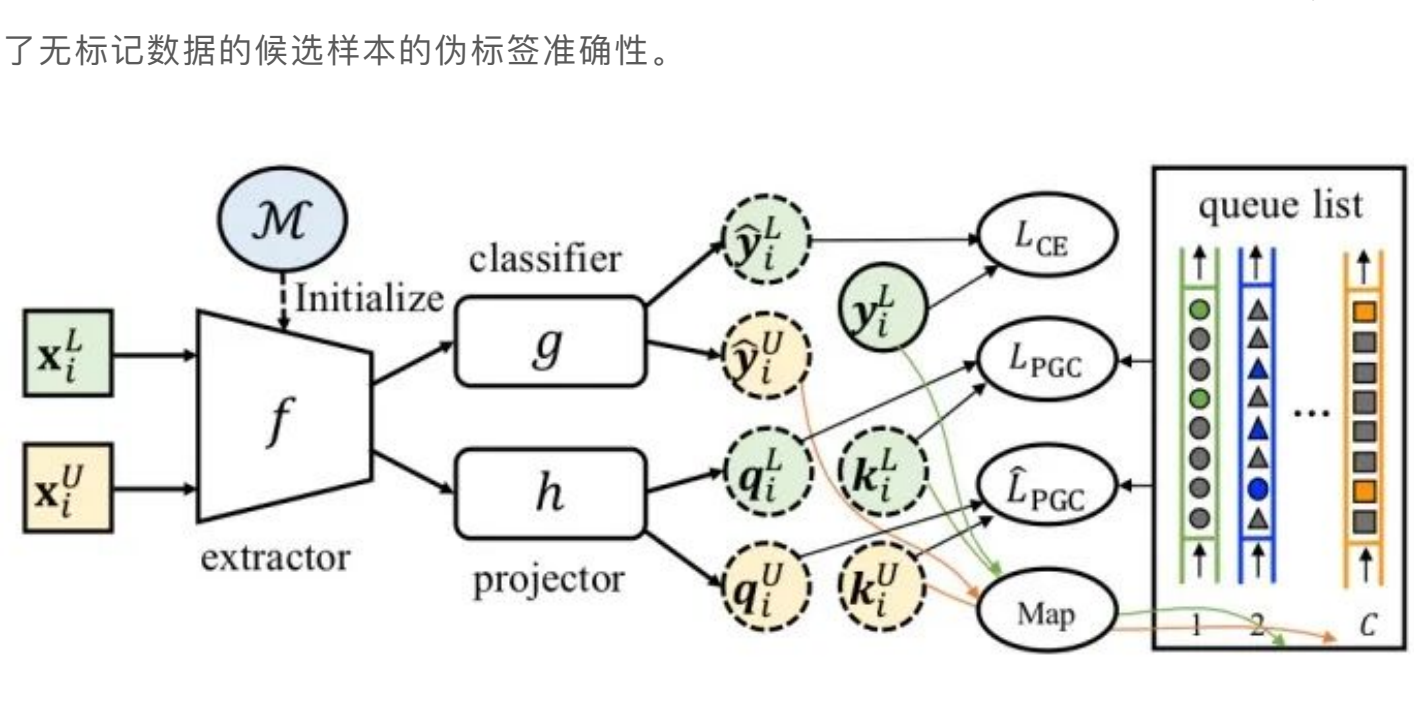


Figure 3. The network architecture of Self-Tuning. The “*Map*” denotes a mapping function which assigns a newly-generated key to the corresponding queue according to its label or pseudo-label.

实验

在实验部分，我们在5种数据集、3种标记数据比例和4种预训练模型下，测试了Self-Tuning的效果，同时与5种主流迁移学习方法、6种主流半监督学习方法以及他们的至强组合进行了充分的对比。

迁移学习的Benchmark

我们首先在迁移学习的常用数据集CUB-200-2011，Stanford Cas和FGVC Aircraft下进行实验，将标记数据的比例依次设置为15%，30%和50%，采用ResNet-50作为预训练模型。结果显示，Self-Tuning大幅领先于现有方法，例如，**在标签比例为15%的Stanford-Cars数据集上，Self-Tuning的测试精度比fine-tuning几乎提高了一倍。**

Table 1. Classification accuracy (%) ↑ of Self-Tuning and various baselines on standard TL benchmarks (ResNet-50 pre-trained).

Dataset	Type	Method	Label Proportion			
			15%	30%	50%	100%
CUB-200-2011	TL	Fine-Tuning (baseline)	45.25±0.12	59.68±0.21	70.12±0.29	78.01±0.16
		L^2-SP (Li et al., 2018)	45.08±0.19	57.78±0.24	69.47±0.29	78.44±0.17
		DELTA (Li et al., 2019)	46.83±0.21	60.37±0.25	71.38±0.20	78.63±0.18
		BSS (Chen et al., 2019)	47.74±0.23	63.38±0.29	72.56±0.17	78.85±0.31
		Co-Tuning (You et al., 2020)	52.58±0.53	66.47±0.17	74.64±0.36	81.24±0.14
	SSL	Π-model (Laine & Aila, 2017)	45.20±0.23	56.20±0.29	64.07±0.32	–
		Pseudo-Labeling (Lee, 2013)	45.33±0.24	62.02±0.31	72.30±0.29	–
		Mean Teacher (Tarvainen & Valpola, 2017)	53.26±0.19	66.66±0.20	74.37±0.30	–
		UDA (Xie et al., 2020)	46.90±0.31	61.16±0.35	71.86±0.43	–
		FixMatch (Sohn et al., 2020)	44.06±0.23	63.54±0.18	75.96±0.29	–
		SimCLRv2 (Chen et al., 2020b)	45.74±0.15	62.70±0.24	71.01±0.34	–
	Combine	Co-Tuning + Pseudo-Labeling	54.11±0.24	68.07±0.32	75.94±0.34	–
		Co-Tuning + Mean Teacher	57.92±0.18	67.98±0.25	72.82±0.29	–
		Co-Tuning + FixMatch	46.81±0.21	58.88±0.23	73.07±0.29	–
		Self-Tuning (ours)	**64.17**±0.47	**75.13**±0.35	**80.22**±0.36	**83.95**±0.18
Stanford Cars	TL	Fine-Tuning (baseline)	36.77±0.12	60.63±0.18	75.10±0.21	87.20±0.19
		L^2-SP (Li et al., 2018)	36.10±0.30	60.30±0.28	75.48±0.22	86.58±0.26
		DELTA (Li et al., 2019)	39.37±0.34	63.28±0.27	76.53±0.24	86.32±0.20
		BSS (Chen et al., 2019)	40.57±0.12	64.13±0.18	76.78±0.21	87.63±0.27
		Co-Tuning (You et al., 2020)	46.02±0.18	69.09±0.10	80.66±0.25	89.53±0.09
	SSL	Π-model (Laine & Aila, 2017)	45.19±0.21	57.29±0.26	64.18±0.29	–
		Pseudo-Labeling (Lee, 2013)	40.93±0.23	67.02±0.19	78.71±0.30	–
		Mean Teacher (Tarvainen & Valpola, 2017)	54.28±0.14	66.02±0.21	74.24±0.23	–
		UDA (Xie et al., 2020)	39.90±0.43	64.16±0.40	71.86±0.56	–
		FixMatch (Sohn et al., 2020)	49.86±0.27	77.54±0.29	84.78±0.33	–
		SimCLRv2 (Chen et al., 2020b)	45.74±0.16	61.70±0.18	77.49±0.24	–
	Combine	Co-Tuning + Pseudo-Labeling	50.16±0.23	73.76±0.26	83.33±0.34	–
		Co-Tuning + Mean Teacher	52.98±0.19	71.42±0.24	75.38±0.29	–
		Co-Tuning + FixMatch	42.34±0.19	73.24±0.25	83.13±0.34	–
		Self-Tuning (ours)	**72.50**±0.45	**83.58**±0.28	**88.11**±0.29	**90.67**±0.23
FGVC Aircraft	TL	Fine-tuning (baseline)	39.57±0.20	57.46±0.12	67.93±0.28	81.13±0.21
		L^2-SP (Li et al., 2018)	39.27±0.24	57.12±0.27	67.46±0.26	80.98±0.29
		DELTA (Li et al., 2019)	42.16±0.21	58.60±0.29	68.51±0.25	80.44±0.20
		BSS (Chen et al., 2019)	40.41±0.12	59.23±0.31	69.19±0.13	81.48±0.18
		Co-Tuning (You et al., 2020)	44.09±0.67	61.65±0.32	72.73±0.08	83.87±0.09
	SSL	Π-model (Laine & Aila, 2017)	37.32±0.25	58.49±0.26	65.63±0.36	–
		Pseudo-Labeling (Lee, 2013)	46.83±0.30	62.77±0.31	73.21±0.39	–
		Mean Teacher (Tarvainen & Valpola, 2017)	51.59±0.23	71.62±0.29	80.31±0.32	–
		UDA (Xie et al., 2020)	43.96±0.45	64.17±0.49	67.42±0.53	–
		FixMatch (Sohn et al., 2020)	55.53±0.26	71.35±0.35	78.34±0.43	–
		SimCLRv2 (Chen et al., 2020b)	40.78±0.21	59.03±0.29	68.54±0.30	–
	Combine	Co-Tuning + Pseudo-Labeling	49.15±0.32	65.62±0.34	74.57±0.40	–
		Co-Tuning + Mean Teacher	51.46±0.25	64.30±0.28	70.85±0.35	–
		Co-Tuning + FixMatch	53.74±0.23	69.91±0.26	80.02±0.32	–
		Self-Tuning (ours)	**64.11**±0.32	**76.03**±0.25	**81.22**±0.29	**84.28**±0.14

半监督学习的Benchmark

在半监督学习的主流数据集CIFAR-100、CIFAR-10、SVHN和STL-10中，我们采用了类别数最多、最困难的CIFAR-100数据集。由于在ImageNet上预训练的WRN-28-8模型尚未公开，我们采用了参数少得多的EfficientNet-B2模型。实验结果表明，预训练模型的引入对于半监督学习有如虎添翼的效果。同时，由于采用了对伪标签依赖更小的PGC损失函数，Self-Tuning充分挖掘了预训练模型、标记数据和无标记数据的所有信息，**在各种实验设定下均取得了state-of-the-art的测试准确率**。

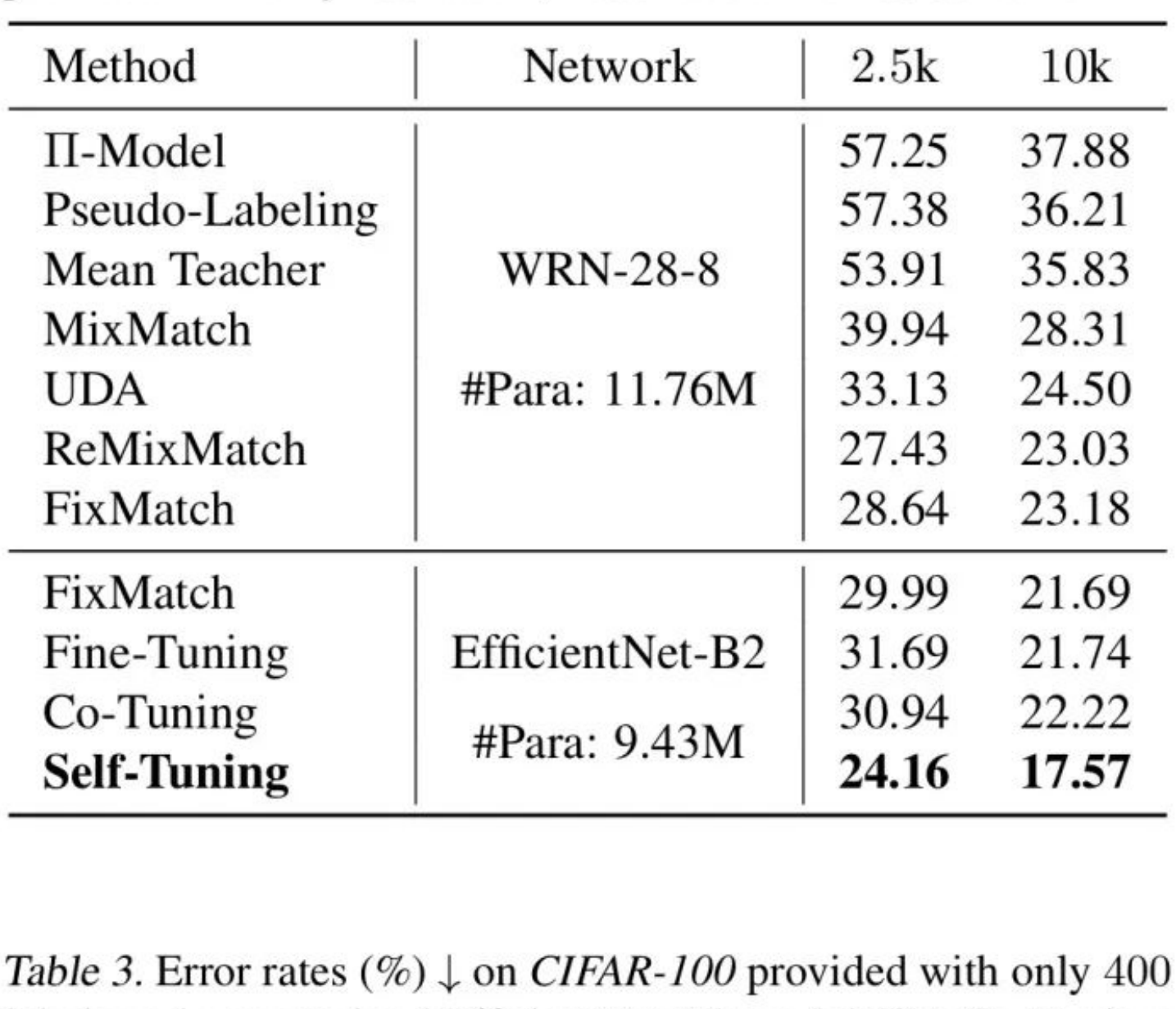


Table 2. Error rates (%) ↓ on standard SSL benchmark: *CIFAR-100* provided with only 400 labels, 2500 labels and 10000 labels.

Method	Network	2.5k	10k
Π-Model	WRN-28-8 #Para: 11.76M	57.25	37.88
Pseudo-Labeling		57.38	36.21
Mean Teacher		53.91	35.83
MixMatch		39.94	28.31
UDA		33.13	24.50
ReMixMatch		27.43	23.03
FixMatch		28.64	23.18
FixMatch	EfficientNet-B2 #Para: 9.43M	29.99	21.69
Fine-Tuning		31.69	21.74
Co-Tuning		30.94	22.22
Self-Tuning		**24.16**	**17.57**

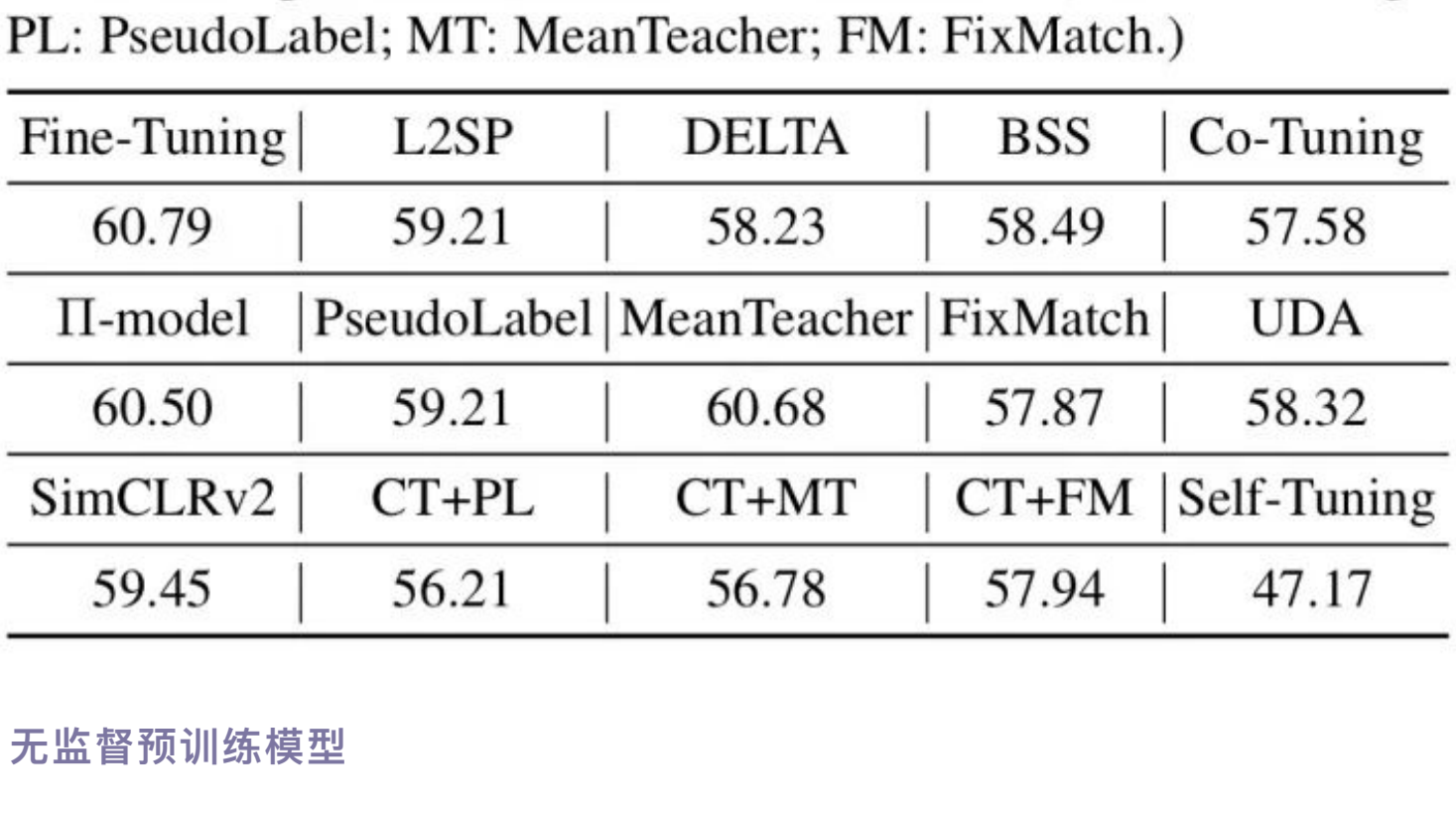


Table 3. Error rates (%) ↓ on *CIFAR-100* provided with only 400 labels and a pre-trained EfficientNet-B2 model (CT: Co-Tuning; PL: PseudoLabel; MT: MeanTeacher; FM: FixMatch.)

Fine-Tuning	L2SP	DELTA	BSS	Co-Tuning
60.79	59.21	58.23	58.49	57.58
Π-model	PseudoLabel	MeanTeacher	FixMatch	UDA
60.50	59.21	60.68	57.87	58.32
SimCLRv2	CT+PL	CT+MT	CT+FM	Self-Tuning
59.45	56.21	56.78	57.94	47.17

无监督预训练模型

为了证明Self-Tuning可以拓展到无监督预训练模型中，我们做了MoCov2迁移到CUB-200的实验。**无论是每类4个样本还是每类25个样本的实验设定，Self-Tuning相较于迁移学习和半监督学习的方法都有明显提升。**

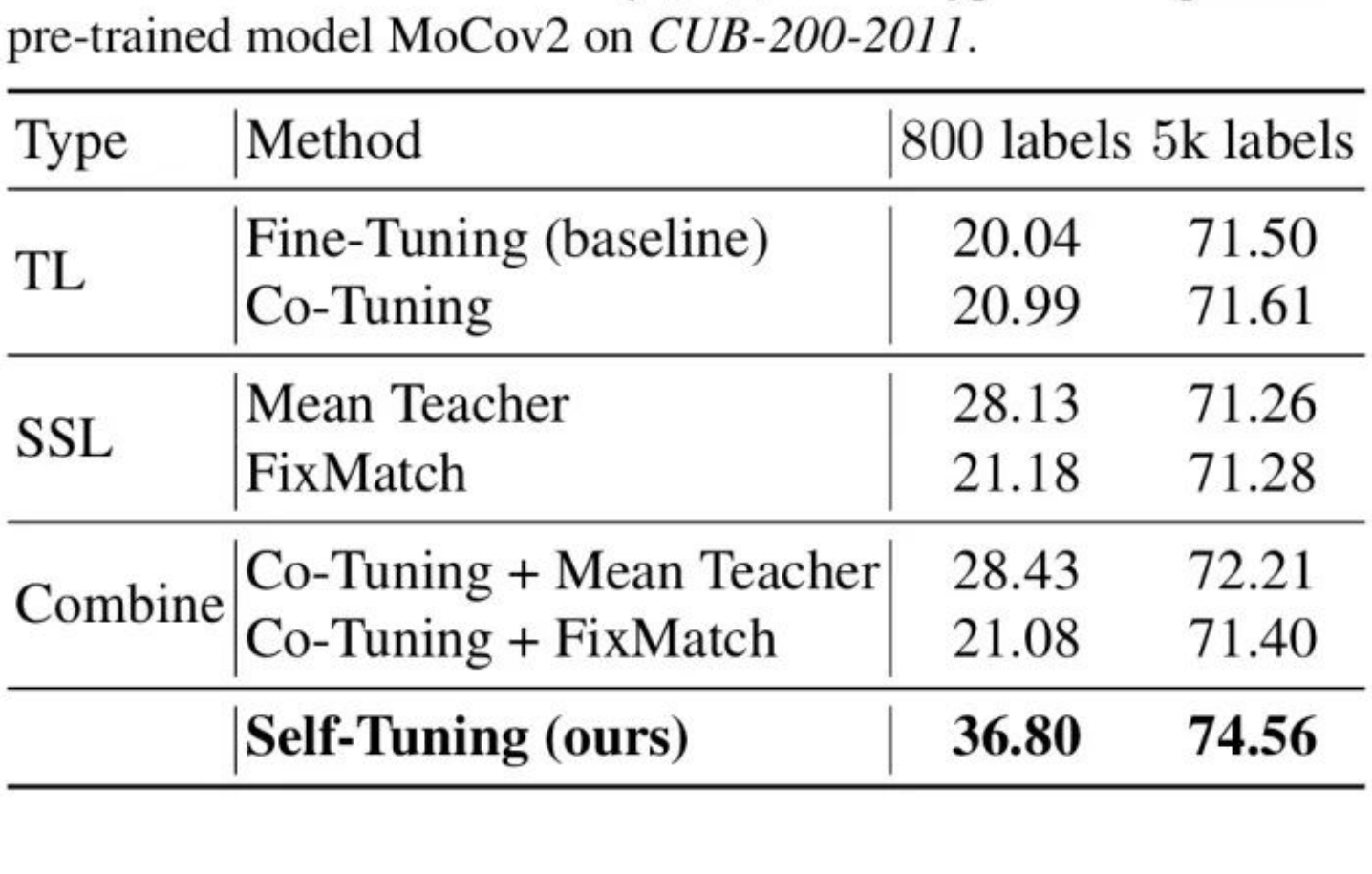


Table 4. Classification accuracy (%) ↑ with a typical unsupervised pre-trained model MoCov2 on *CUB-200-2011*.

Type	Method	800 labels	5k labels
TL	Fine-Tuning (baseline)	20.04	71.50
	Co-Tuning	20.99	71.61
SSL	Mean Teacher	28.13	71.26
	FixMatch	21.18	71.28
Combine	Co-Tuning + Mean Teacher	28.43	72.21
	Co-Tuning + FixMatch	21.08	71.40
	Self-Tuning (ours)	**36.80**	**74.56**

命名实体识别

为了证明Self-Tuning可以拓展到自然语言处理(NLP)的任务中，我们在一个英语命名实体识别数据集CoNLL 2003上进行了实验。按照Co-Tuning的实验设定，我们采用掩蔽语言建模的BERT作为预训练模型。以命名实体的F1得分作为度量指标的话，fine-tuning的F1得分为90.81，BSS、L2-SP和Co-Tuning分别达到90.85、91.02和91.27，**而Self-Tuning取得了明显更高的94.53的F1得分，初步证明了Self-Tuning在NLP领域的强大潜力**。更加详尽的NLP实验，会在未来的期刊版本中进行拓展。

消融实验

在消融实验部分，我们从两个不同的角度进行了对比。首先是**损失函数**，PGC损失函数比Cross-Entropy和Contrastive Learning的损失函数有明显提升。其次是**信息的探索方式**，无论是去掉标记数据还是无标记数据上的PGC损失函数，抑或在标记数据和无标记数据间设置单独的负样本队列，都不及Self-Tuning所提的“一体化”信息探索。

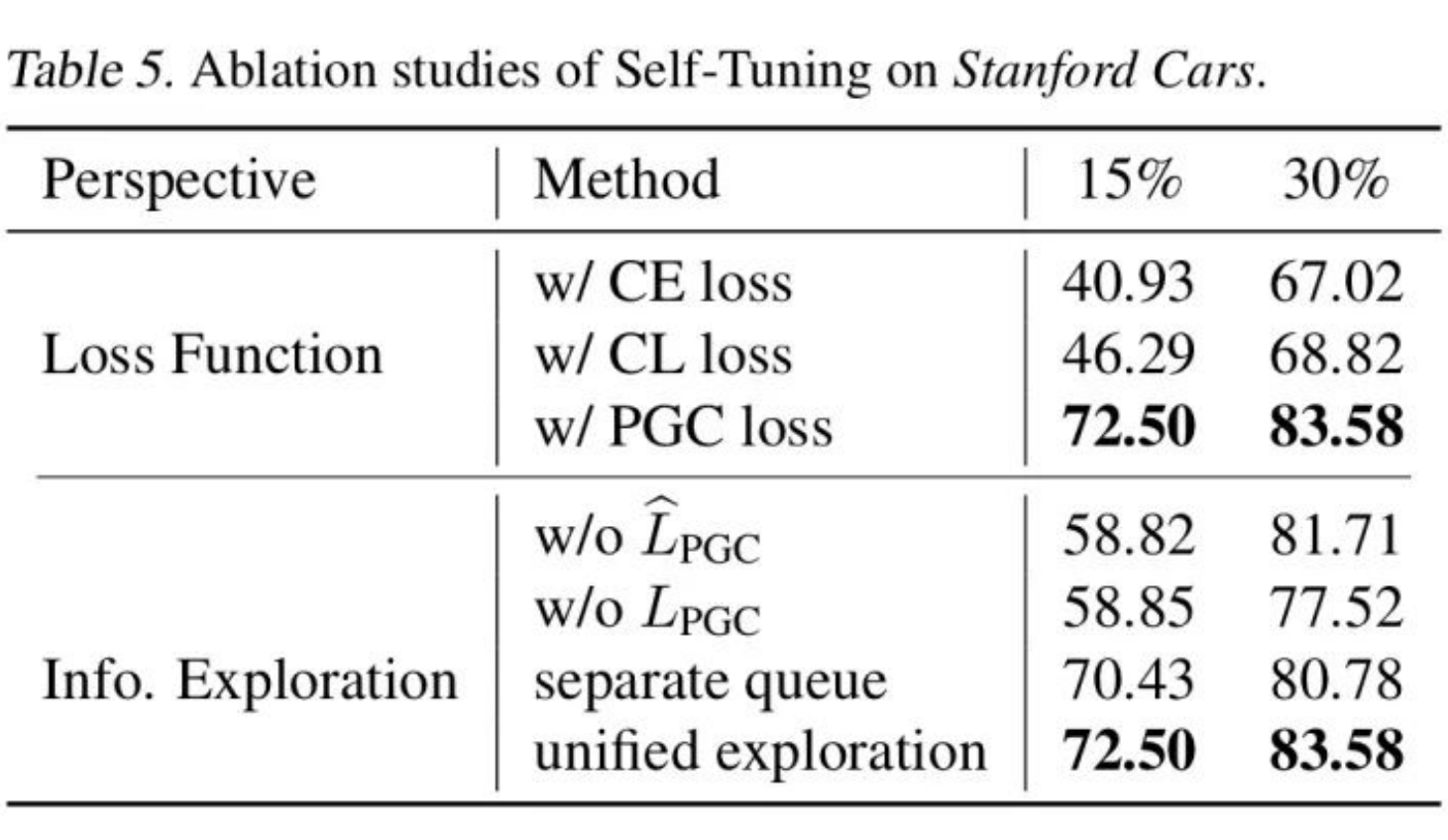


Table 5. Ablation studies of Self-Tuning on *Stanford Cars*.

Perspective	Method	15%	30%
Loss Function	w/ CE loss	40.93	67.02
	w/ CL loss	46.29	68.82
	w/ PGC loss	**72.50**	**83.58**
Info. Exploration	w/o $\widehat{L}_{\text{PGC}}$	58.82	81.71
	w/o L_{PGC}	58.85	77.52
	separate queue	70.43	80.78
	unified exploration	**72.50**	**83.58**

展望

在深度学习社区中，如何减少对标记数据的需求是一个至关重要的问题。考虑到迁移学习和半监督学习的普通实践中只关注预训练模型或无标记数据的不足，本文提出了一种新的数据高效的深度学习机制，可以充分发挥预训练模型和无标记数据的优势。这一机制可能是迁移学习在工业界最为现实的落地场景，值得我们继续大力研究。另一方面，我们提出的Self-Tuning方法简单通用，是迁移学习、半监督学习和对比学习等领域的核心思想的集大成者，可以提高对伪标签的容忍度。对于其他需要用到伪标签的场景，应该也有一定的借鉴价值。