Self-Tuning for Data-Efficient Deep Learning

Ximei Wang*, Jinghan Gao*, Mingsheng Long (✉), and Jianmin Wang

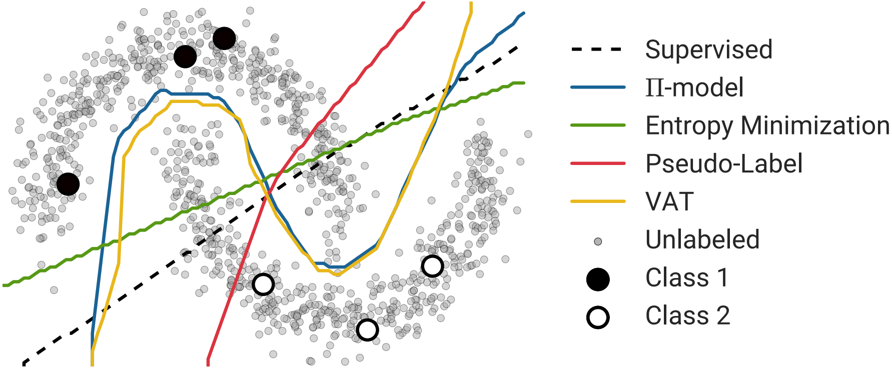
School of Software, BNRist, Tsinghua University

wxm17@mails.tsinghua.edu.cn, https://wxm17.github.io/

International Conference on Machine Learning (ICML), 2021

Semi-supervised Learning (SSL)

Simultaneously exploring both labeled and unlabeled data [1]

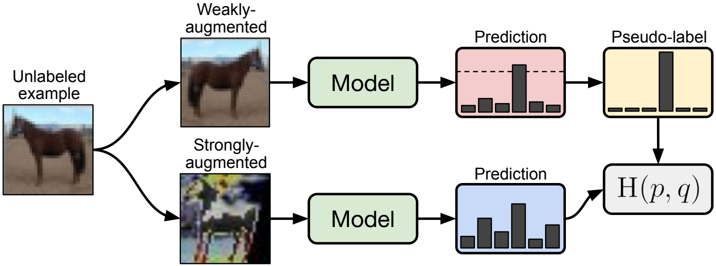


[1] *Oliver et al. Realistic Evaluation of Deep Semi-Supervised Learning Algorithms. NeurIPS 2018.*

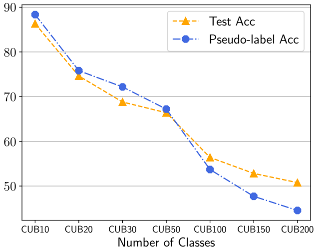
Delve into a State-of-the-art SSL Method: FixMatch[2]

Main Idea: Use the model's predictions on *weakly-augmented* unlabeled images to generate pseudo-labels for *strongly-augmented* versions of the same images.

Confirmation Bias: The performance of a student is restricted by the teacher when learning from inaccurate pseudo-labels.



(a) Diagram of FixMatch

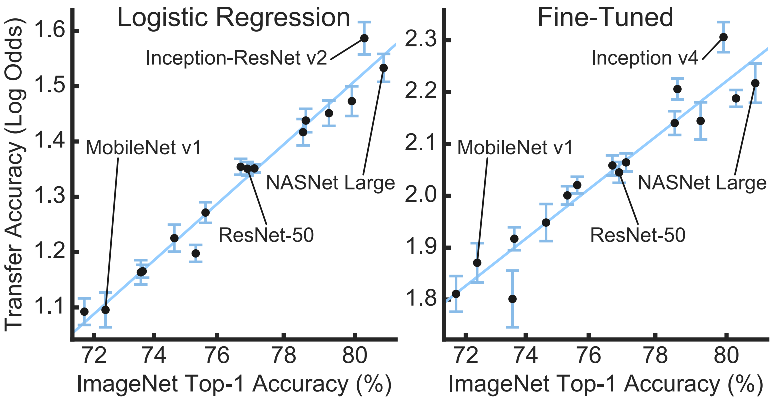


(b) Accuracy w.r.t label size

[2] *Sohn et al. Realistic Evaluation of Deep Semi-Supervised Learning Algorithms. NeurIPS 2018.*

Transfer Learning (TL)

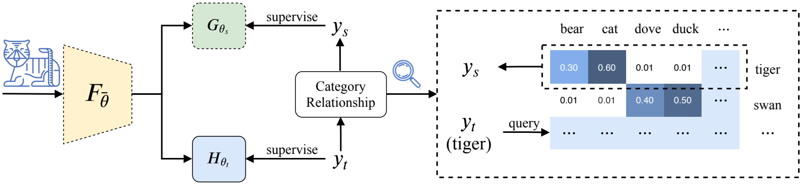
Fine-tuning a pre-trained model to the target data [3]



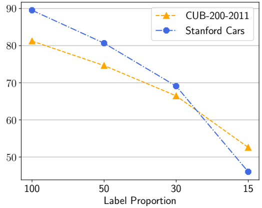
[3] *Kornblith et al. Do Better ImageNet Models Transfer Better? CVPR 2019.*

Delve into a State-of-the-art TL Method: Co-Tuning[4]

Main Idea: Learn the *relationship* between source categories and target categories from the pre-trained model with calibrated prediction to fully transfer pre-trained models.
Model Shift: The fine-tuned model shifts towards the limited labeled data, without exploring the intrinsic structure of unlabeled data.



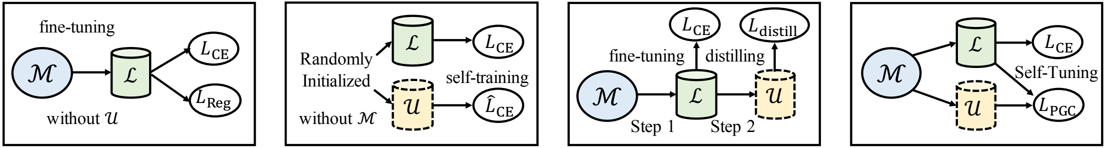
(a) Diagram of Co-Tuning



(b) Acc w.r.t label ratio

[4] *You et al. Co-Tuning for Transfer Learning. NeurIPS 2020.*

Data-Efficient Deep Learning



(a) Transfer Learning (b) SSL (c) SimCLRv2 (d) **Self-Tuning** (ours)

Figure: 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 $\mathcal{M}$.

How to Tackle Confirmation Bias?

- The Devil Lies in Cross-Entropy Loss
- Contrastive Learning Loss Underutilizes Labels

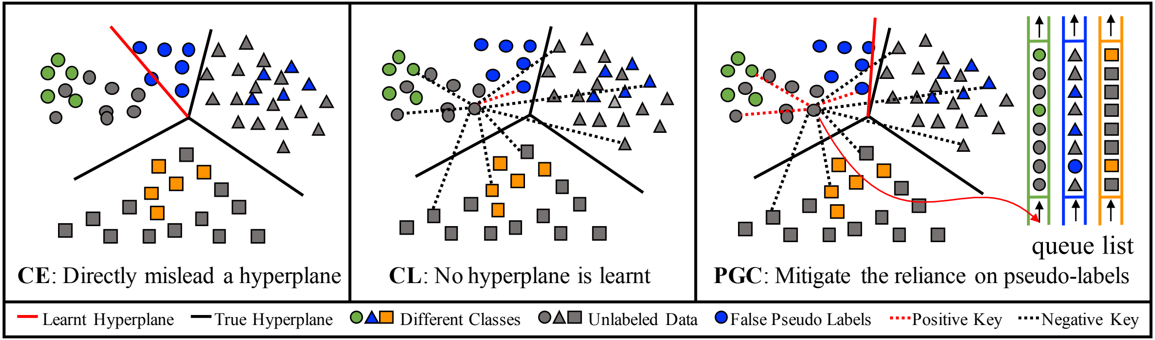


Figure: Conceptual comparison of various loss functions: (a) **CE**: cross-entropy loss will be easily misled by false pseudo-labels; (b) **CL**: contrastive learning loss underutilizes labels and pseudo-labels; (c) **PGC**: Pseudo Group Contrast mechanism to mitigate confirmation bias.

From Contrastive Learning to Pseudo Group Contrast (PGC)

- **Contrastive Learning**: maximizes the similarity between the query q with its corresponding positive key k_0 (a differently augmented view of the *same* data example)

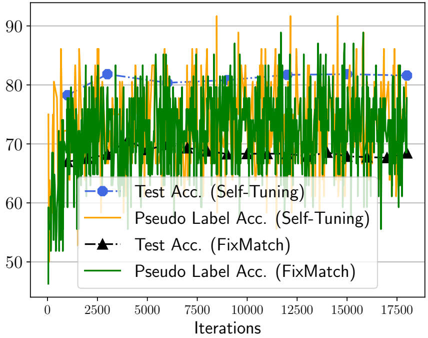
$$L_{\mathrm{CL}} = -\log \frac{\exp(\mathrm{q} \cdot \mathrm{k}_0/\tau)}{\exp(\mathrm{q} \cdot \mathrm{k}_0/\tau) + \sum_{d=1}^{D} \exp(\mathrm{q} \cdot \mathrm{k}_d/\tau)}, \tag{1}$$

- **Pseudo Group Contrast**: introduces *a group of positive keys in the same pseudo-class* to contrast with all negative keys from other pseudo-classes.

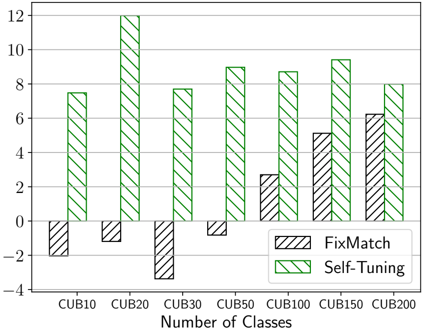
$$\widehat{L}_{\mathrm{PGC}} = -\frac{1}{D+1} \sum_{d=0}^{D} \log \frac{\exp(\mathrm{q} \cdot \mathrm{k}_d^{\widehat{y}}/\tau)}{\exp(\mathrm{q} \cdot \mathrm{k}_0^{\widehat{y}}/\tau) + \sum_{c=1}^{\{1,2,\cdots,C\}} \sum_{j=1}^{D} \exp(\mathrm{q} \cdot \mathrm{k}_j^{c}/\tau)}, \tag{2}$$

Why can PGC boost the tolerance to false labels?

- The *softmax* function generates a predicted probability vector with a sum of 1. Positive keys $\{k_0^{\widehat{y}}, k_1^{\widehat{y}}, k_2^{\widehat{y}}, \cdots, k_D^{\widehat{y}}\}$ from the same pseudo-class will compete with each other.
- If some pseudo-labels in the positive group are wrong, those keys with true pseudo-labels will win, since their representations are more similar to the query, compared to false ones.



(a) Training Process on *CUB30*



(b) $\mathrm{Acc}_{\mathrm{test}} - \mathrm{Acc}_{\mathrm{pseudo\_labels}}$

Model Shift: Unifying and Sharing

- **A unified form to fully exploit $\mathcal{M}$, $\mathcal{L}$ and $\mathcal{U}$**
- **A shared queue list across $\mathcal{L}$ and $\mathcal{U}$**

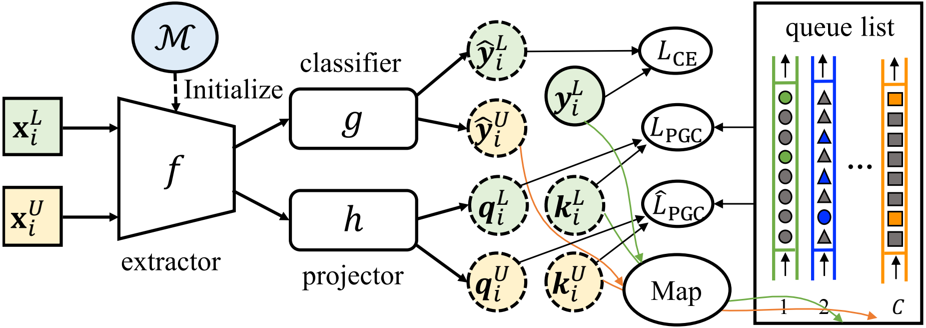


Figure: The network architecture of Self-Tuning. The "*Map*" denotes a mapping function which assigns a newly-generated key to the corresesoping queue according to its label or pseudo-label.

Experiments and Results

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

Summary

- A new setup named data-efficient deep learning to unleash the power of both transfer learning and semi-supervised learning.
- To tackle model shift and confirmation bias problems, we propose *Self-Tuning* to unify the exploration of labeled and unlabeled data and the transfer of a pre-trained model.
- A general Pseudo Group Contrast mechanism to mitigate the reliance on pseudo-labels and boost the tolerance to false labels.
- Comprehensive experiments demonstrate that *Self-Tuning* outperforms its SSL and TL counterparts on five tasks by sharp margins.
- Code will be available at @ `github.com/thuml/Self-Tuning`