

Co-Tuning for Transfer Learning


Kaichao You, Zhi Kou, Mingsheng Long, Jianmin Wang

Tsinghua University

Neural Information Processing Systems (NeurIPS 2020)



Transfer Learning

 PyTorch

Get Started Ecosystem Mobile Blog **Tutorials** Docs Resources Github

Tutorials > Transfer Learning for Computer Vision Tutorial

1.6.0


Search Tutorials

PyTorch Recipes

See All Recipes

Run in Google Colab Download Notebook View on GitHub

TRANSFER LEARNING FOR COMPUTER VISION TUTORIAL

 TensorFlow

Install Learn API Resources Community Why TensorFlow

TensorFlow Core

Overview **Tutorials** Guide TF 1

TensorFlow tutorials

Quickstart for beginners

Quickstart for experts

BEGINNER

ML basics with Keras

Load and preprocess data

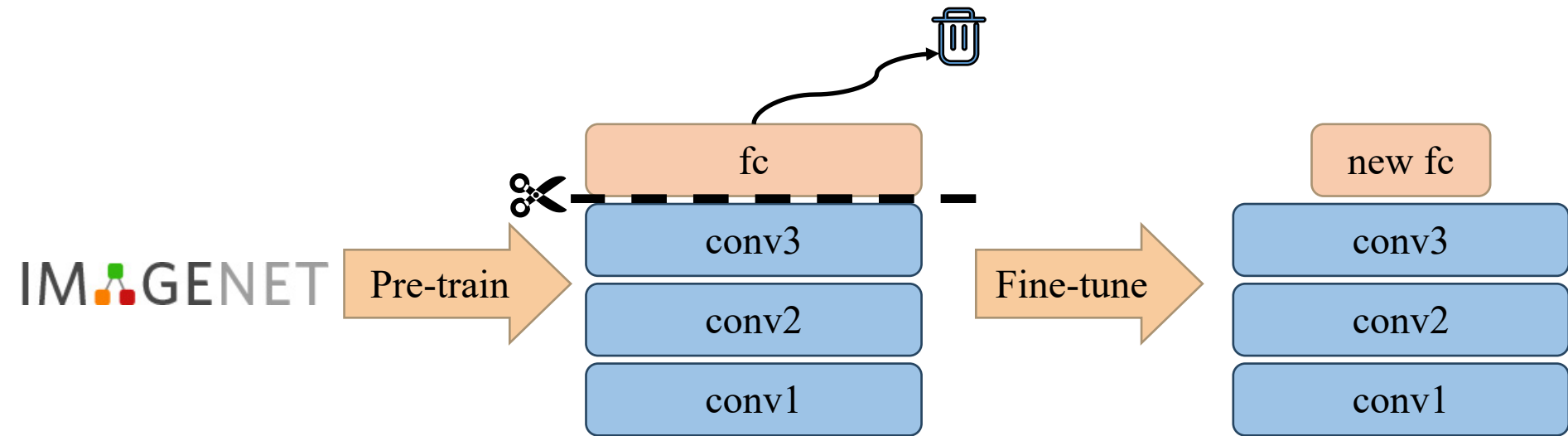
TensorFlow > Learn > TensorFlow Core > Tutorials

Transfer learning and fine-tuning

Run in Google Colab View source on GitHub Download notebook



Status Quo



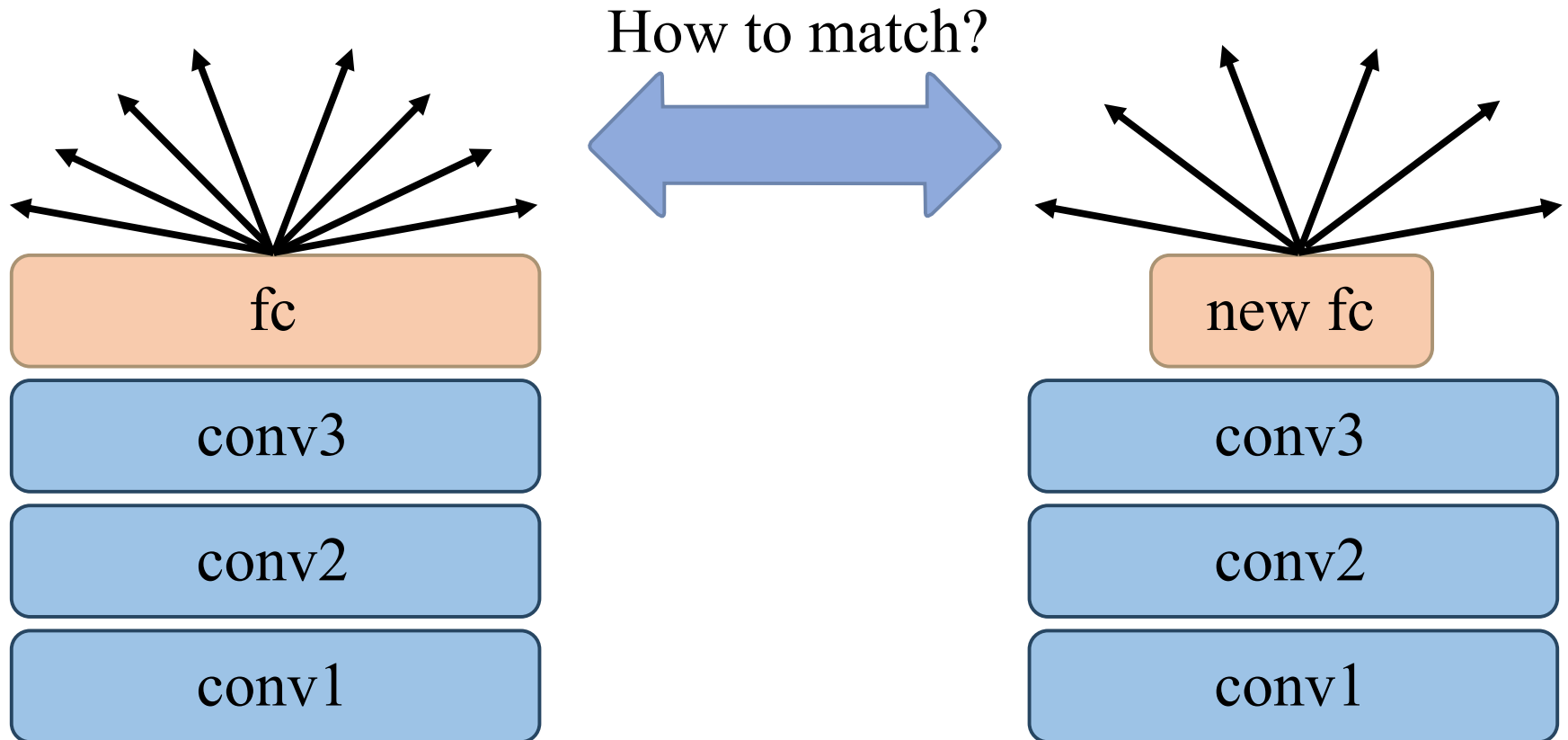
Parameter count in popular pre-trained models from torchvision and transformers.

Pre-trained model	ResNet-50	DenseNet-121	Inception-V3	BERT-base
Task-specific parameters / Million	2.0	1.0	2.0	22.9
Total parameters / Million	25.6	8.0	27.2	108.9
Percentage / %	7.8	12.5	7.4	21.0

Can we reuse task-specific pre-trained layer(s)?






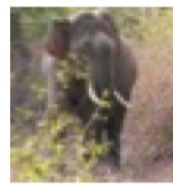















Problem





Co-Tuning Solution

Learn the category relationship $p(y_s | y_t)$

COCO categories	ImageNet categories				
elephant	<i>indian elephant</i>	<i>african elephant</i>	<i>tusker</i>	<i>others</i>	
					
=	+	+	+		
					
	30%	26%	25%	19%	
donut	<i>bagel</i>	<i>pretzel</i>	<i>bakery</i>	<i>others</i>	
					
=	+	+	+		
					
	21%	11%	7%	61%	



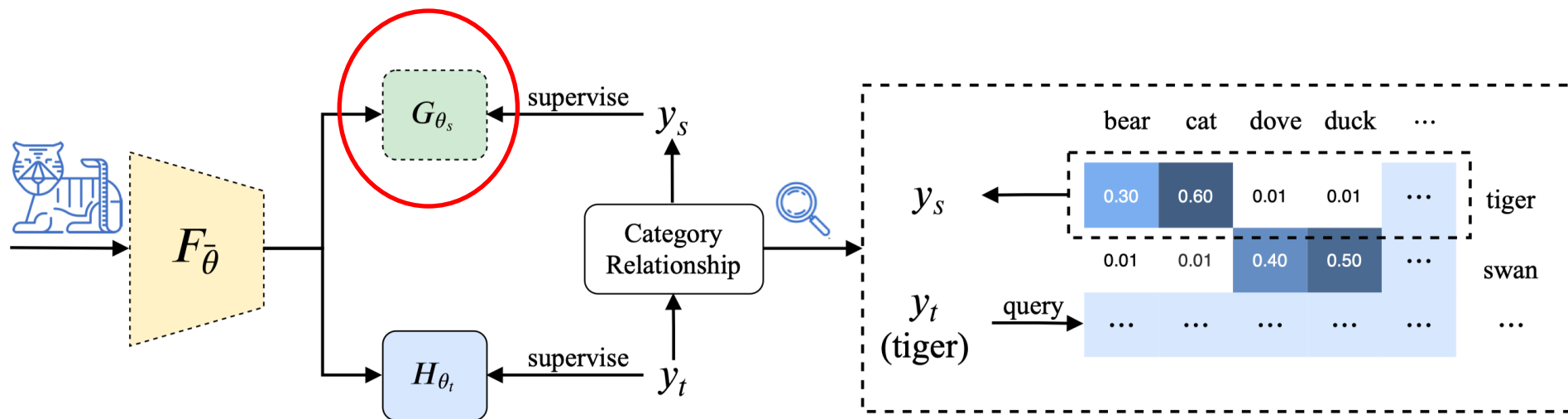
Co-Tuning Solution

Learn the category relationship $p(y_s | y_t)$

- Direct approach
 - $f_0(x) \approx p(y_s | x)$
 - average source predictions for each target category
$$p(y_s | y_t = y) \approx |\mathcal{D}_t^y|^{-1} \sum_{(x, y_t) \in \mathcal{D}_t^y} f_0(x), \quad \mathcal{D}_t^y = \{(x, y_t) \in \mathcal{D}_t | y_t = y\}$$
- Reserse approach
 - learn the mapping $y_s \rightarrow y_t$ from $(f_0(x_t), y_t)$ pairs, which is $p(y_t | y_s)$
 - compute y_t marginal from target labeled data
 - recover $p(y_s | y_t)$ from $p(y_t | y_s)$ and y_t by Bayes's rule
- Calibration (optional)
 - calibrate pre-trained models if source validation data is available
 - can be transformed into a simple convex optimization problem
$$t^* = \arg \min_{t > 0} \sum_{i=1}^m \text{cross_entropy}(\text{softmax}(f(x^i)/t), y^i)$$



Co-Tuning Solution



- Pre-trained models are fully transferred
- No additional inference cost



Results

Table 2: Classification accuracy in medium-scale classification datasets (Pre-trained ResNet-50).

Dataset	Method	Sampling Rates			
		15%	30%	50%	100%
CUB-200-2011	Fine-tune (baseline)	45.25 \pm 0.12	59.68 \pm 0.21	70.12 \pm 0.29	78.01 \pm 0.16
	L ² -SP (Li et al., 2018)	45.08 \pm 0.19	57.78 \pm 0.24	69.47 \pm 0.29	78.44 \pm 0.17
	DELTA (Li et al., 2019)	46.83 \pm 0.21	60.37 \pm 0.25	71.38 \pm 0.20	78.63 \pm 0.18
	BSS (Chen et al., 2019)	47.74 \pm 0.23	63.38 \pm 0.29	72.56 \pm 0.17	78.85 \pm 0.31
	Co-Tuning	52.58 \pm 0.53	66.47 \pm 0.17	74.64 \pm 0.36	81.24 \pm 0.14
Stanford Cars	Fine-tune (baseline)	36.77 \pm 0.12	60.63 \pm 0.18	75.10 \pm 0.21	87.20 \pm 0.19
	L ² -SP (Li et al., 2018)	36.10 \pm 0.30	60.30 \pm 0.28	75.48 \pm 0.22	86.58 \pm 0.26
	DELTA (Li et al., 2019)	39.37 \pm 0.34	63.28 \pm 0.27	76.53 \pm 0.24	86.32 \pm 0.20
	BSS (Chen et al., 2019)	40.57 \pm 0.12	64.13 \pm 0.18	76.78 \pm 0.21	87.63 \pm 0.27
	Co-Tuning	46.02 \pm 0.18	69.09 \pm 0.10	80.66 \pm 0.25	89.53 \pm 0.09
FGVC Aircraft	Fine-tune (baseline)	39.57 \pm 0.20	57.46 \pm 0.12	67.93 \pm 0.28	81.13 \pm 0.21
	L ² -SP (Li et al., 2018)	39.27 \pm 0.24	57.12 \pm 0.27	67.46 \pm 0.26	80.98 \pm 0.29
	DELTA (Li et al., 2019)	42.16 \pm 0.21	58.60 \pm 0.29	68.51 \pm 0.25	80.44 \pm 0.20
	BSS (Chen et al., 2019)	40.41 \pm 0.12	59.23 \pm 0.31	69.19 \pm 0.13	81.48 \pm 0.18
	Co-Tuning	44.09 \pm 0.67	61.65 \pm 0.32	72.73 \pm 0.08	83.87 \pm 0.09

Table 3: Classification accuracy in large-scale COCO-70 dataset (Pre-trained DenseNet-121).

Method	Sampling Rates			
	15%	30%	50%	100%
Fine-tune (baseline)	76.60 \pm 0.04	80.15 \pm 0.25	82.50 \pm 0.43	84.41 \pm 0.22
L ² -SP (Li et al., 2018)	77.53 \pm 0.47	80.67 \pm 0.29	83.07 \pm 0.39	84.78 \pm 0.16
DELTA (Li et al., 2019)	76.94 \pm 0.37	79.72 \pm 0.24	82.00 \pm 0.52	84.66 \pm 0.08
BSS (Chen et al., 2019)	77.39 \pm 0.15	80.74 \pm 0.22	82.75 \pm 0.59	84.71 \pm 0.13
Co-Tuning	77.64 \pm 0.23	81.19 \pm 0.18	83.43 \pm 0.22	85.65 \pm 0.11

- Works across different pre-trained models and dataset sizes

Thanks
