Distilling Cross-Task Knowledge via Relationship Matching

Han-Jia Ye, Su Lu, De-Chuan Zhan   Nanjing University

Highlights

- *We propose REFILLED approach to distill knowledge from a **cross-task teacher** trained on non-overlapping classes.*
- *We emphasize that the **comparison ability between instances** is an essential factor to relate two domains.*
- *State-of-the-art experimental results under 3 settings.*

Deep Networks for Classification

- Input: training dataset $\mathcal{D} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^N$ where $\mathbf{x}_i \in \mathbb{R}^D$ and $\mathbf{y}_i \in \{0,1\}^C$.
- Objective: $\min_f \sum_{i=1}^N \ell(f(\mathbf{x}_i), \mathbf{y}_i)$ where $\ell(\cdot,\cdot)$ is a loss function such as cross-entropy.
- Output: a deep network $f(x): \mathbb{R}^D \mapsto \{0,1\}^C$ which can be decomposed into a feature extractor $\phi: \mathbb{R}^D \mapsto \mathbb{R}^d$ and a linear classifier $W \in \mathbb{R}^{d\times C}$.

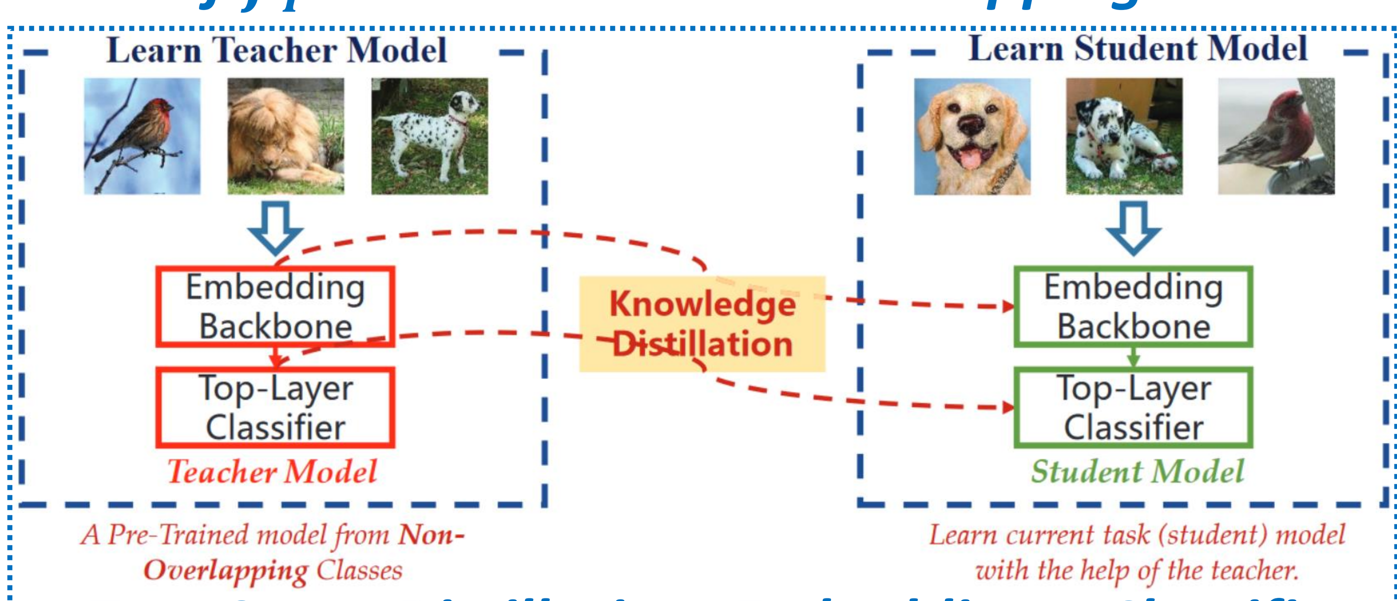
Knowledge Distillation

- Aside from training dataset, an extra model well-trained on the same task f_T (a.k.a teacher) is given.
- Distill "dark knowledge" from f_T to improve the training efficacy of f_S (a.k.a student).
- Let $s_\tau(f(\mathbf{x}_i)) = softmax(f(\mathbf{x}_i)/\tau)$, we solve

$$\min_{f_S} \sum_{i=1}^N \ell(f_S(\mathbf{x}_i), \mathbf{y}_i) + \lambda \mathcal{R}(s_\tau(f_T(\mathbf{x}_i)), s_\tau(f_S(\mathbf{x}_i))),$$

where $\mathcal{R}(\cdot,\cdot)$ measures the difference between two distributions, e.g., Kullback-Leibler divergence.

What if f_T is trained on non-overlapping classes?

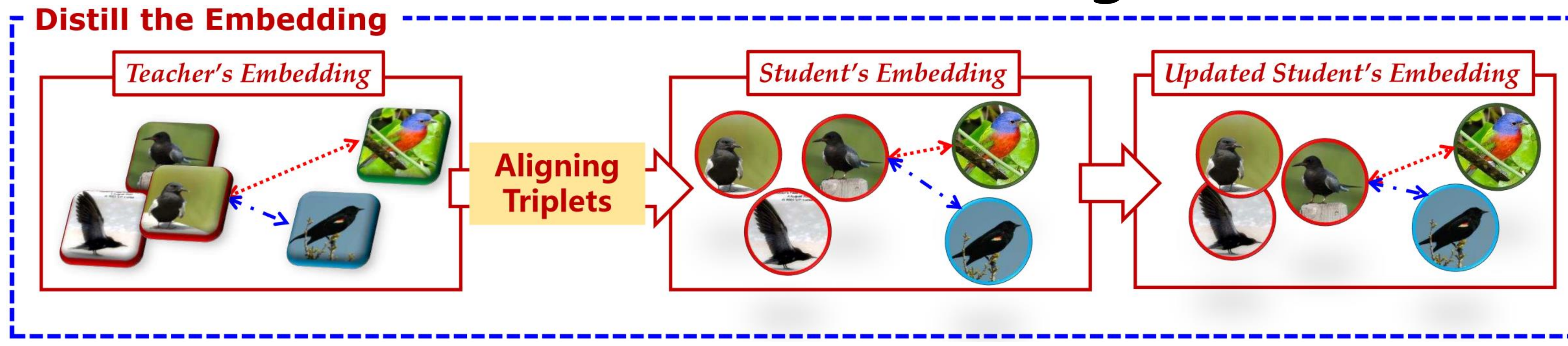


Two-Stage Distillation: Embedding + Classifier

Algorithm 1 The Flow of REFILLED.

Require: Pre-trained Teacher's Embedding ϕ_T.
 Distill the Embedding:
 for all Iter = 1,...,MaxIter **do**
  Sample a mini-batch $\{(\mathbf{x}_i, \mathbf{y}_i)\}$.
  Generate triplets $\{(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k)\}$ with student's embeddings $\{\phi_S(\mathbf{x})\}$.
  Compute probability of triplets $p_{ijk}(\phi_T)$ as Eq. 4.
  Optimizing ϕ_S by aligning triplets in Eq. 5.
 end for
 Distill the Classifier:
 Initialize f_S with ϕ_S.
 Optimizing f_S with Eq. 9.

Distill the Embedding



Main Idea: Triplet Alignment by Stochastic Triplet Probability

- A triplet $(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k)$ contains an anchor $\mathbf{x}_i$, its neighbor $\mathbf{x}_j$, and its impostor $\mathbf{x}_k$.
- Based on embedding ϕ, the stochastic triplet probability encodes how much the anchor is closer to its target neighbor than its impostor.

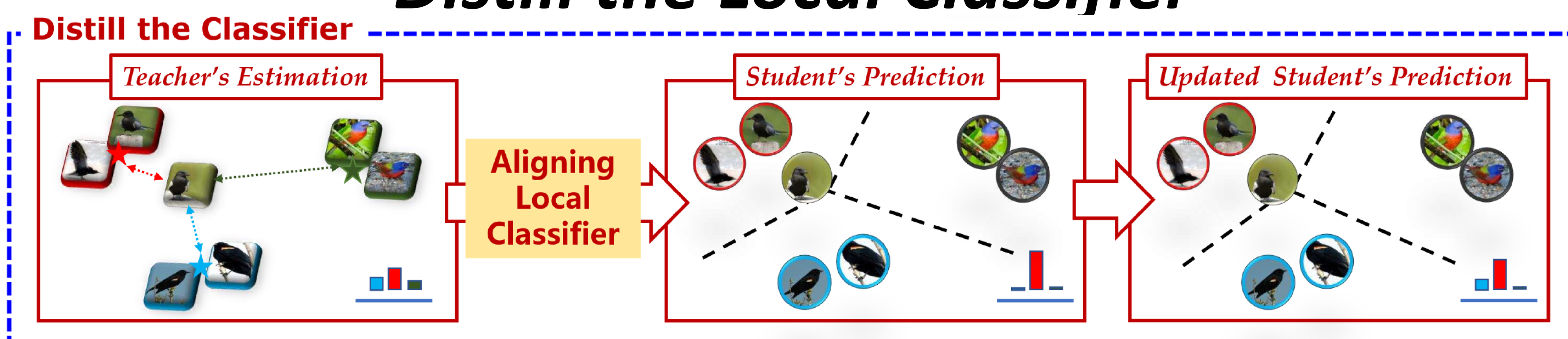
$$p_{ijk}(\phi) = \frac{\exp(-\mathrm{Dist}_\phi(\mathbf{x}_i, \mathbf{x}_j)/\tau)}{\exp(-\mathrm{Dist}_\phi(\mathbf{x}_i, \mathbf{x}_j)/\tau) + \exp(-\mathrm{Dist}_\phi(\mathbf{x}_i, \mathbf{x}_k)/\tau)}$$

$$\mathrm{Dist}_\phi(\mathbf{x}_i, \mathbf{x}_j) = \|\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)\|_2$$

- Construct a Bernoulli distribution $\mathcal{P}_{ijk}(\phi) = [p_{ijk}(\phi), 1 - p_{ijk}(\phi)]$.
- Minimize the KL-divergence between $\mathcal{P}_{ijk}(\phi_T)$ and $\mathcal{P}_{ijk}(\phi_S)$ over generated triplets.

$$\min_{\phi_S} \sum_{ijk} \mathbf{KL}(\mathcal{P}_{ijk}(\phi_T) || \mathcal{P}_{ijk}(\phi_S))$$

Distill the Local Classifier



Main Idea: Distillation from Mini-Batch Nearest Class Mean Classifier

- We construct an embedding based local classifier. Denote $X \in \mathbb{R}^{N\times D}$ and $Y \in \{0,1\}^{N\times C}$ as the instances and one-hot labels in a sampled mini-batch. We compute the embedding center of each class as P, where $\oslash$ denotes element-wise division.

$$P = \mathrm{diag}(\mathbf{1} \oslash (Y^T\mathbf{1}))Y^T\phi_T(X) \in \mathbb{R}^{C\times d}$$

- The label of an instance in the batch can be determined by teacher model:

$$p_{\phi_T}(\mathbf{y}_i|\mathbf{x}_i) = \mathrm{softmax}(-\|\phi_T(\mathbf{x}_i) - \mathbf{p}_c\|_2^2/\tau), c = 1, \ldots, C$$

- Knowledge distillation on classes in the sampled mini-batch rather than all C classes.

$$\min_{f_S} \sum_{i=1}^N \ell(f_S(\mathbf{x}_i), \mathbf{y}_i) + \lambda KL(p_{\phi_T}(\mathbf{y}_i|\mathbf{x}_i) || s_\tau(f_S(\mathbf{x}_i)))$$

State-of-the-art results

Cross-Task Knowledge Distillation

- Dataset: CUB-200 split into two parts
- Model: MobileNet with different channels

	Easy				Hard			
Channel	1	0.75	0.5	0.25	1	0.75	0.5	0.25
Teacher	1NN: 49.23, LR: 56.77, FT: 66.94				1NN: 45.31, LR: 53.82, FT: 65.72			
Student	70.04	68.13	66.44	64.63	71.25	67.56	66.85	64.48
RKD [13]	71.10	68.81	67.15	64.28	70.83	68.8	67.44	63.97
Vanilla	71.62	70.27	70.15	66.75	71.90	69.14	68.91	65.38
LKD	71.93	70.73	70.88	67.41	72.53	70.01	69.50	66.42
REFILLED	72.48	71.04	71.35	67.87	73.38	70.42	69.77	67.10

- ***REFILLED outperforms several baseline methods and some other comparison methods.***
- ***1NN: Nearest Neighbor using teacher model***
- ***LR: Logistic Regression using teacher model***
- ***FT: Fine-tuning the teacher model***

We propose REFILLED, a cross-task knowledge distillation method by reusing the comparison ability of teacher model, which works well in cross-task KD, standard KD, and middle-shot learning.
Code: https://github.com/njulus/REFILLED

Standard Knowledge Distillation

(depth, width)	(40, 2)	(16, 2)	(40, 1)	(16, 1)
Teacher	74.44			
Student	74.44	70.15	68.97	65.44
KD [20]	75.47	71.87	70.46	66.54
FitNet [43]	74.29	70.89	68.66	65.38
AT [67]	74.76	71.06	69.85	65.31
NST [23]	74.81	71.19	68.00	64.95
VID-I [2]	75.25	73.31	71.51	66.32
KD+VID-I [2]	76.11	73.69	72.16	67.19
RKD [38]	76.62	72.56	72.18	65.22
REFILLED	**77.49**	**74.01**	**72.72**	**67.56**

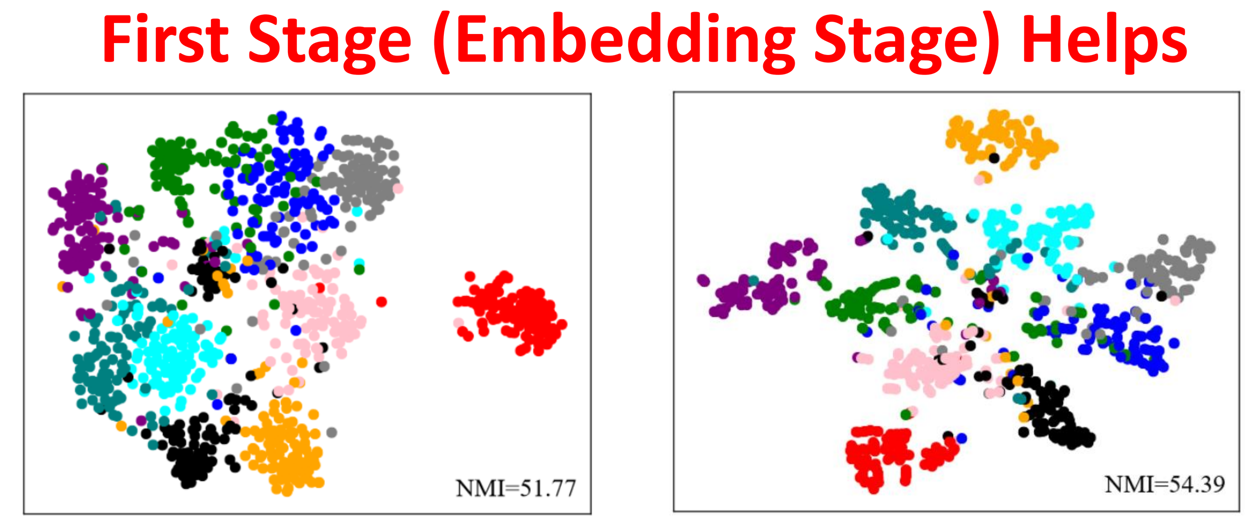
CIFAR-100, WideResNet -> WideResNet

Width Multiplier	1	0.75	0.5	0.25
Teacher	75.36			
Student	75.36	74.87	72.41	69.72
KD [20]	77.61	76.02	74.24	72.03
FitNet [43]	75.10	75.03	72.17	69.09
AT [67]	76.22	76.10	73.70	70.74
NST [23]	76.91	77.05	74.03	71.54
KD+VID-I [2]	77.03	76.91	75.62	72.23
RKD [38]	77.72	76.80	74.99	72.55
REFILLED	**78.95**	**78.01**	**76.11**	**73.42**

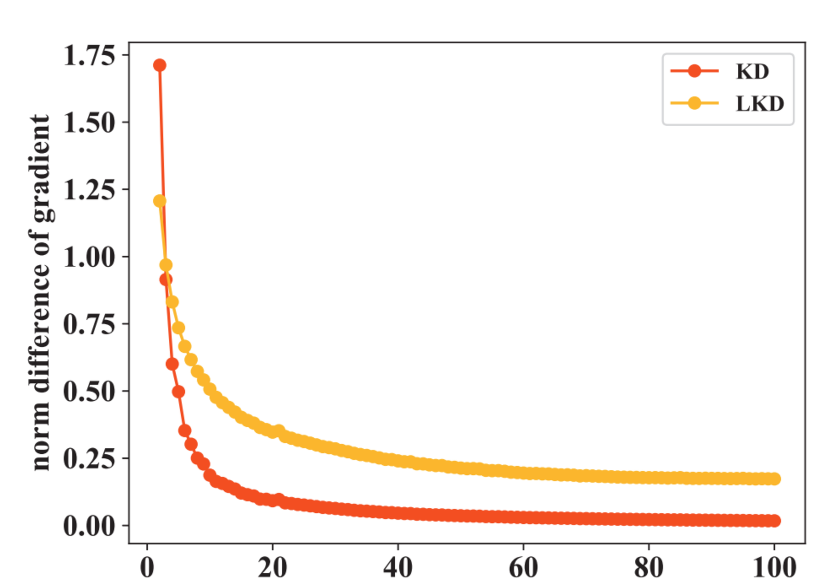
CUB-200, MobileNet -> MobileNet

- ***REFILLED works well in standard knowledge distillation and middle-shot learning.***

First Stage (Embedding Stage) Helps



Second Stage (Classifier Stage) Helps



When the number of classes in a task grows, the gradient norm difference decreases fast. But the decrease is mitigated when using LKD loss.

Middle-Shot Learning

Tasks	1-Shot 5-Way	5-Shot 5-Way	10-Shot 5-Way	30-Shot 5-Way
1NN	49.73	63.11	66.56	69.80
SVM	51.61	69.17	74.24	77.87
Fine-Tune	45.89	68.61	74.95	78.62
MAML [3]	48.70	63.11	-	-
ProtoNet [20]	51.79	70.38	74.42	78.10
FEAT [26]	55.15	71.61	74.86	78.84
REFILLED[1]	54.82	71.97	76.42	80.33
REFILLED[2]	53.44	70.60	75.37	78.94

REFILLED1: ResNet backbone; REFILLED2: ConvNet backbone