Transferability-Guided Cross-Domain Cross-Task Transfer Learning

Yang Tan, Yang Li*, Shao-Lun Huang, Xiao-Ping Zhang

Tsinghua-Berkeley Shenzhen Institute, Tsinghua University

* Corresponding author

MIIS 2022

TBSI

Abstract

To solve the *"when and what to transfer"* problem under the challenging cross-domain cross-task transfer learning setting, we propose a unified framework namely **OTCE** (**O**ptimal **T**ransport based **C**onditional **E**ntropy), which can not only quantitatively evaluate that *how much the source task (model) benefits the learning of the target task*, but also serves as a loss function to *optimize the source representations to become more transferable* to the target task.

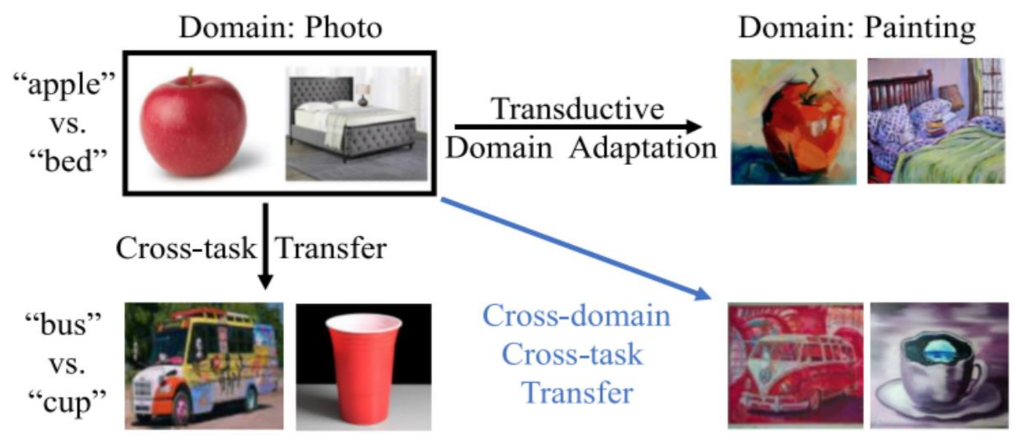
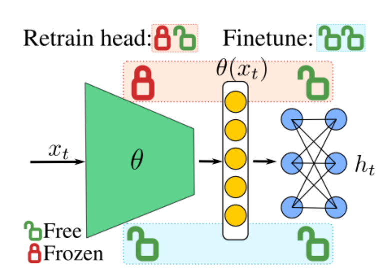


Figure 1: Illustration of different transfer learning settings.

When to Transfer (Transferability Estimation) [1]

Transferability Definition

- Source data $D_s = \{(x_s^i, y_s^i)\}_{i=1}^m \sim P_s(x,y)$
- Target data $D_t = \{(x_t^i, y_t^i)\}_{i=1}^n \sim P_t(x,y)$
- Pretrained source model (θ_s, h_s) — Head classifier: h_s; Feature extractor: θ_s



- Empirical Transferability

$$\mathrm{Trf}(S \to T) = \begin{cases} \mathbb{E}\left[\log P(y_t|x_t;\theta_s,h_t)\right] & \text{(Retrain head)} \\ \mathbb{E}\left[\log P(y_t|x_t;\theta_t,h_t)\right] & \text{(Finetune)} \end{cases}$$

- Our OTCE metric aims to efficiently approximate the empirical transferability.

OTCE Metric

- Embed samples into feature space $z_s^i = \theta_s(x_s^i)$ $\quad z_t^i = \theta_s(x_t^i)$
- Solve the Optimal Transport problem to obtain an optimal coupling matrix π^*,

$$OT(Z_s, Z_t) \triangleq \min_{\pi \in \mathcal{P}(Z_s,Z_t)} \sum_{i,j=1}^{m,n} c(z_s^i, z_t^j)\pi_{ij} - \lambda H(\pi)$$

where $c(\cdot,\cdot) = ||\cdot - \cdot||_2^2$ is the cost metric, and $H(\pi) = -\sum_{i=1}^m \sum_{j=1}^n \pi_{ij} \log \pi_{ij}$ is the entropic regularizer.

- Compute $\hat{P}(y_s, y_t)$ and $\hat{P}(y_s)$ as:

$$\hat{P}(y_s, y_t) = \sum_{i,j:y_s^i = y_s, y_t^j = y_t} \pi_{ij}^*, \quad \hat{P}(y_s) = \sum_{y_t \in \mathcal{Y}_t} \hat{P}(y_s, y_t)$$

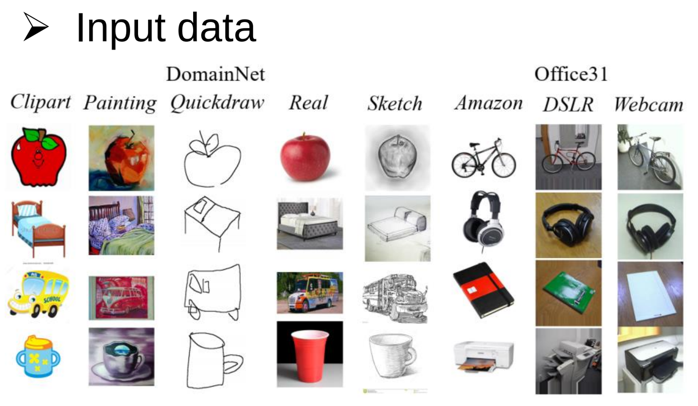
- **OTCE** score is computed as the Negative Conditional Entropy

$$\mathrm{OTCE} = -H_{\pi^*}(Y_t|Y_s) = \sum_{y_t \in \mathcal{Y}_t} \sum_{y_s \in \mathcal{Y}_s} \hat{P}(y_s, y_t) \log \frac{\hat{P}(y_s, y_t)}{\hat{P}(y_s)}$$

- $-H(Y_t|Y_s)$ lower bounds the log likelihood of the target classifier using the source feature extractor.[3]

Experiments

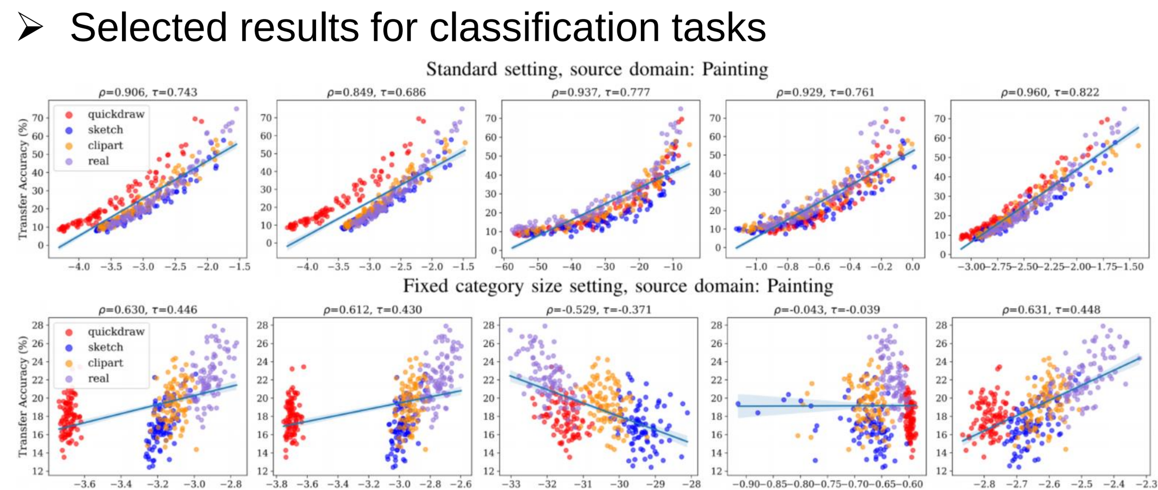
- Input data



Data for classification tasks

Data for segmentation tasks

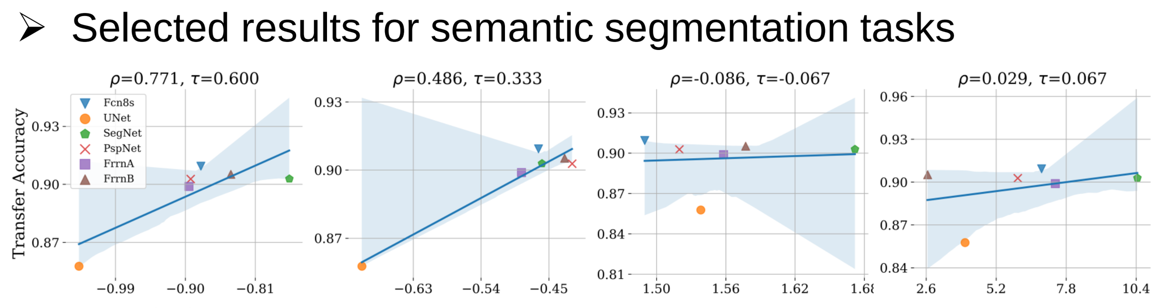
- Selected results for classification tasks



Correlation between transfer accuracy and transferability scores.

Each point denotes a classification task.

- Selected results for semantic segmentation tasks



Each point denotes a model architecture.

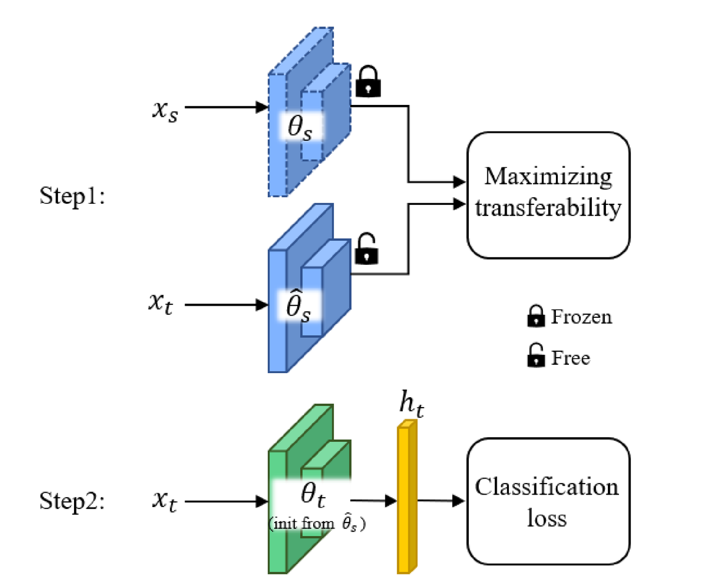
What to Transfer (Learn Transferable Representation) [2]

OTCE-based Finetune

- Maximizing OTCE score will lead to a more transferable feature representation.

$$\hat{\theta}_s^* = \arg\min_{\hat{\theta}_s} H_{\pi^*}(Y_t|Y_s)$$

$$= -\arg\min_{\hat{\theta}_s} \sum_{y_t \in \mathcal{Y}_t} \sum_{y_s \in \mathcal{Y}_s} \hat{P}(y_s, y_t) \log \frac{\hat{P}(y_s, y_t)}{\hat{P}(y_s)}$$



Algorithm 6.1 OTCE-based finetune

Require: source dataset $D_s = \{(x_s^i, y_s^i)\}_{i=1}^m$; target dataset $D_t = \{(x_t^i, y_t^i)\}_{i=1}^n$; source feature extractor θ_s

Initialize $\hat{\theta}_s = \theta_s$
while sampling mini-batches within one epoch **do**
  Generate mini-batch $B_s = \{(\theta_s(x_s^i), y_s^i)\}_i^M$
  Generate mini-batch $B_t = \{(\hat{\theta}_s(x_t^i), y_t^i)\}_i^N$
  Update $\hat{\theta}_s$ via maximizing F-OTCE(B_s, B_t)
end while
Initialize $\theta_t = \hat{\theta}_s$
Randomly initialize h_t
while θ_t, h_t not converge **do**
  Update θ_t, h_t using equation (6.2)
end while

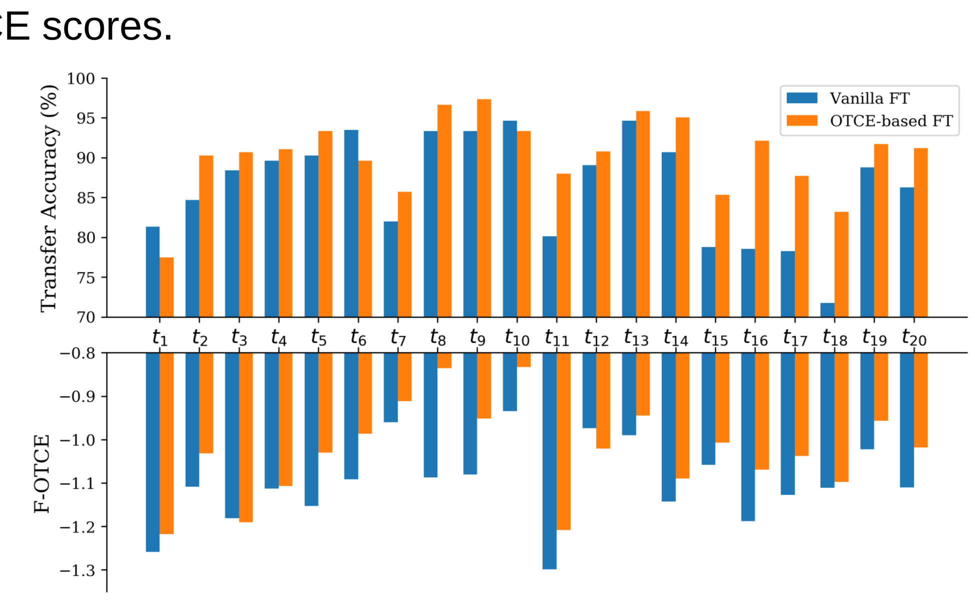
Note: F-OTCE and OTCE are substitutes for each other.

Experiments

- Cross-domain few-shot transfer learning experiments.

Model	Method	MNIST → Omniglot	Caltech101 → MiniImageNet
FewshotNet	MAML [32]	$88.60 \pm 1.14\%$	$28.23 \pm 0.44\%$
	MatchingNet [30]	$87.92 \pm 1.10\%$	$44.75 \pm 1.30\%$
	ProtoNet [33]	$83.11 \pm 1.34\%$	$50.40 \pm 1.35\%$
	RelationNet [31]	$69.35 \pm 1.62\%$	$29.55 \pm 0.61\%$
	Vanilla finetune	$91.30 \pm 0.95\%$	$49.49 \pm 1.27\%$
	OTCE-based finetune	$\mathbf{92.32 \pm 0.87\%}$	$\mathbf{51.36 \pm 1.33\%}$
LeNet	Vanilla finetune	$86.11 \pm 1.10\%$	-
	OTCE-based finetune	$\mathbf{90.52 \pm 0.94\%}$	-
ResNet-18	Vanilla finetune	-	$48.48 \pm 1.39\%$
	OTCE-based finetune	-	$\mathbf{50.02 \pm 1.34\%}$

- Visual comparison: the optimized models exhibit both higher transfer accuracy and OTCE scores.



Discussion

Conclusions

- OTCE evaluates transferability for both image classification and semantic segmentation tasks.
- It also helps the source model to be more transferable to the target task.
- It has been applied in source model (task) selection, multi-source feature fusion, etc.

Future works

- Deeply understand the theoretical explanations of our method.
- Explore more applications in transfer learning, e.g., domain generalization.

Reference

[1] Y. Tan, Y. Li, SL. Huang. OTCE: A Transferability Metric for Cross-Domain Cross-Task Representations. In CVPR 2021.
[2] Y. Tan, Y. Li, SL. Huang, XP. Zhang. Transferability-Guided Cross-Domain Cross-Task Transfer Learning. In arXiv:2207.05510.
[3] A. Tran, C. Nguyen, T. Hassner. Transferability and Hardness of Supervised Classification Tasks. In ICCV 2019.

Code