TransBoost: Improving The Best ImageNet Performance Using Deep Transduction

Technion Israel Institute of Technology

deci

Omer Belhasin* **Guy Bar-Shalom*** **Ran El-Yaniv**

NEURAL INFORMATION PROCESSING SYSTEMS

Abstract

This paper deals with deep transductive learning, and proposes TransBoost as a procedure for fine-tuning any deep neural model to improve its performance on any (unlabeled) test sample provided at training time. TransBoost is inspired by a large margin principle and is efficient and simple to use. Our method significantly improves the ImageNet classification performance on a wide range of architectures, such as ResNets, MobileNetV3-L, EfficientNetB0, ViT-S, and ConvNext-T, leading to state-of-the-art transductive performance (which significantly outperforms the inductive one). Additionally, we show that TransBoost is effective on a wide variety of image classification datasets. Open source implementation of TransBoost is available at: https://github.com/omerb01/TransBoost.

Problem Formulation: Transduction - not induction

A learner is provided with the following:

- A set of labeled instances, $S_l \triangleq \{(x_1, y_1), (x_2, y_2), \ldots, (x_L, y_L)\}$, sampled i.i.d. from $P(X, Y)$.
- A set of unlabeled instances, $X_u \triangleq \{x_{L+1}, \ldots, x_{L+U}\}$, sampled i.i.d. from $P(X)$.
- A pre-trained model, $f_\theta : \mathcal{X} \to \mathcal{Y}$, trained on the labeled set.

Goal: uncover the labels of X_u. Formally, minimize the true risk over the unlabeled set,

$$E_{f_\theta}(X_U) \triangleq \frac{1}{U} \sum_{i=1}^{U} \ell(f_\theta(x_{L+i}), y_{L+i}) .$$

TransBoost

Overview

Algorithm 1: TransBoost Procedure

Input : Labeled train sample $S_l \triangleq (X_l, Y_l)$, unlabeled test sample X_u, a model f_θ that was pretrained on S_l, a supervised pointwise loss function ℓ, implementations of TransBoost's functions: $\mathcal{S}$, δ and κ. Hyperparameters: Number of epochs E, labeled batch size L', unlabeled batch size U', regularization parameter λ.

for $epoch \leftarrow 1$ **to** E **do**
  `// Sample labeled and unlabeled batches in a cyclical manner`
  for $S_l' \triangleq (X_l', Y_l') \subset S_l$, $X_u' \subset X_u$ **to** $\max\{\lceil \frac{L}{L'} \rceil, \lceil \frac{U}{U'} \rceil\}$ **do**
    `// Prepare random test sample pairs`
    ${X_u'}^{\pi} \leftarrow$ generate a random permutation of X_u'
    `// Approximate the TransBoost loss`
    $L_{\text{TransBoost}} \leftarrow 0$
    for $i \leftarrow 0$ **to** U' **do**
      $L_{\text{TransBoost}} \leftarrow L_{\text{TransBoost}} + \kappa(x_i)\kappa({x_i}^{\pi})\delta(x_i, {x_i}^{\pi})\mathcal{S}(x_i, {x_i}^{\pi})$
    end for
    $L_{\text{TransBoost}} \leftarrow L_{\text{TransBoost}} / \sum_{i=1}^{U'} \delta(x_i, {x_i}^{\pi})$ if $\sum_{i=1}^{U'} \delta(x_i, {x_i}^{\pi}) \neq 0$ else 0
    `// Apply an optimization step`
    $\theta \leftarrow$ optimize $\left[\frac{1}{L'} \sum_{i=1}^{L'} \ell(f_\theta(x_i), y_i) + \lambda \cdot L_{\text{TransBoost}}\right]$
  end for
end for

- TransBoost is a **fine-tuning** procedure that can **improve the performance** of any trained deep neural model with respect to a specific (unlabeled) test set.
- The pretrained model is optimized using a weighted combination:
  - Labeled / **inductive** (standard) loss.
  - Unlabeled / **transduction** loss.

$$\mathcal{L}(X_l, Y_l, X_u \mid f_\theta, \mathcal{S}, \delta, \kappa) \triangleq \underbrace{\frac{1}{L} \sum_{i=1}^{L} \ell(f_\theta(x_i), y_i)}_{\text{labeled/inductive (standard) loss}} + \lambda \cdot \underbrace{\mathcal{L}_{\text{TransBoost}}(X_u \mid f_\theta, \mathcal{S}, \delta, \kappa)}_{\text{unlabeled/transductive loss}}$$

TransBoost Loss Function

Generic Structure

- Our transduction loss function is inspired by **large margin principle** and consists of three components:
  - $\mathcal{S} : X \times X \to \mathcal{R}$ A **similarity** function.
  - $\delta : X \times X \to \{0,1\}$ A **decision** function.
  - $\kappa : X \to \mathcal{R}^+$ A **confidence** function.

$$\mathcal{L}_{\text{TransBoost}}(X_u \mid f_\theta, \mathcal{S}, \delta, \kappa) \triangleq \frac{1}{U_\delta} \sum_{1 \le i < j \le U} \kappa(x_i)\kappa(x_j)\delta(x_i, x_j)\mathcal{S}(x_i, x_j) .$$

Our Implementation

- The **similarity** function is based on the **L2 norm:**

$$\mathcal{S}_f(x_i, x_j) \triangleq \sqrt{2} - ||\hat{p}(x_i \mid f_\theta) - \hat{p}(x_j \mid f_\theta)||_2 .$$

- The **decision** function is based on the **pretrained model** (pseudo labeling):

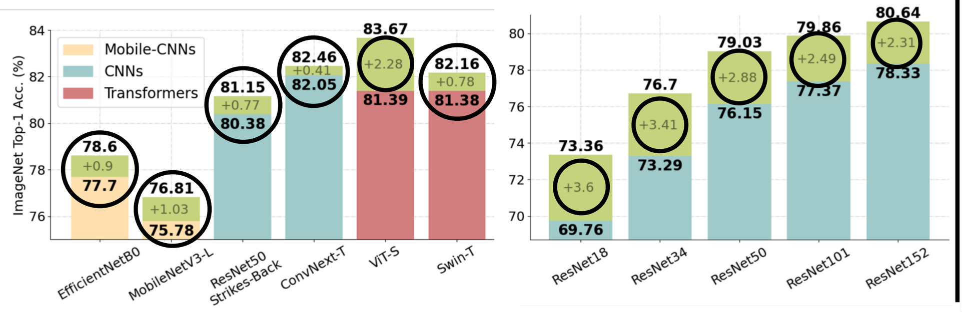
$$\delta_f(x_i, x_j) \triangleq \begin{cases} 1 & f_{\theta_0}(x_i) \neq f_{\theta_0}(x_j) \\ 0 & \text{otherwise} . \end{cases}$$

- The **confidence** function is the **max class probability** (Softmax):

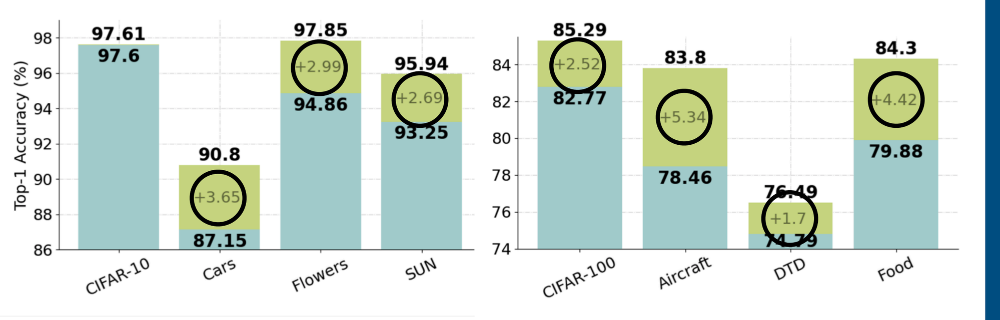
$$\kappa_f(x) \triangleq \max\{\hat{p}_j(x \mid f_{\theta_0})\}_{j=1}^{C} .$$

Results

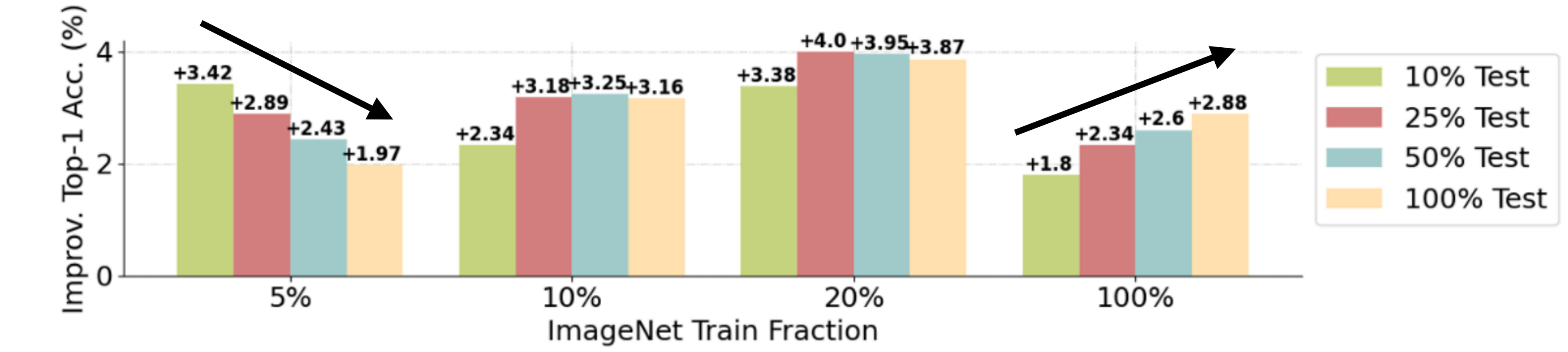
ImageNet:



Additional datasets:



- TransBoost shows consistent and significant improvement over the inductive setting.



- When using a large training set, use a large test set for optimal performance.
- When using a small training set, use a small test set for optimal performance.

Conclusions:

- TransBoost is a powerful transductive fine-tuning procedure that can be efficiently applied to any pertained model.
- It consistently and significantly improves the inductive performance of many architectures and image classification datasets.
- TransBoost exhibits state-of-the-art for transductive classification performance. **If your setting is transductive — use TransBoost!**

TransBoost open source implementation: https://github.com/omerb01/TransBoost.