Fine-Tuning is Fine, if Calibrated

Zheda Mai*[1], **Arpita Chowdhury***[1], **Ping Zhang***[1], Cheng-Hao Tu[1], Hong-You Chen[1], Vardaan Pahuja[1], Tanya Berger-Wolf[1], Song Gao[2], Charles Stewart[3], Yu Su[1], Wei-Lun Chao[1]

[1]The Ohio State University, [2]University of Wisconsin Madison, [3]Rensselaer Polytechnic Institute

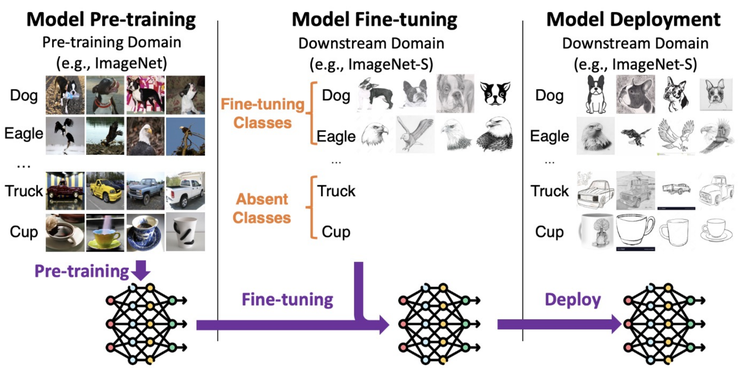
Highlights

- Fine-tuning (FT) a **pre-trained** model that recognized many classes with only a **subset** of classes from a **new domain** significantly **degrades** absent class accuracies.
- Accuracy degradation does **NOT** come from **feature deterioration**, but stems from **biased logits** toward fine-tuned classes.
- Simple **post-processing calibration** restores model's capabilities and reveals improved features.

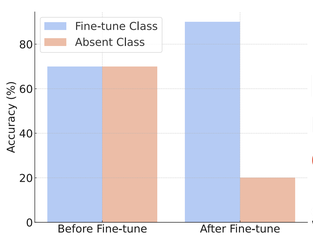
Motivation

Fine-tuning pre-trained models on a **subset** of target classes is often unavoidable, e.g., camera trap, some species may **NOT** appear within the collection time.

Setting



Problem

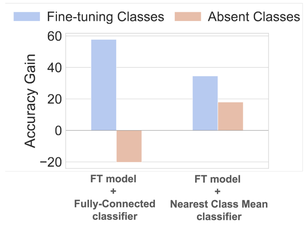


FT with a **subset** of classes from **new** domains causing accuracy **drops** for **absent** classes.

Superficial cause: **Forgetting**!

What are damaged in the fine-tuned (FT) model?

Is the FT feature extractor damaged?



NO! FT with **subset** of classes can **adapt** the **feature extractor** to the new domain and **improve absent** classes.

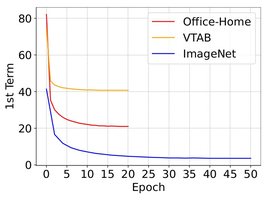
Is the FT classifier damaged?

$$p(c|\boldsymbol{x}) = \frac{\exp\left(\boldsymbol{w}_c^\top f_{\boldsymbol{\theta}}(\boldsymbol{x})\right)}{\sum_{c' \in \mathcal{Y}} \exp\left(\boldsymbol{w}_{c'}^\top f_{\boldsymbol{\theta}}(\boldsymbol{x})\right)} = \frac{z_c(\boldsymbol{x})}{\sum_{c' \in (\mathcal{S} \cup \mathcal{U})} z_{c'}(\boldsymbol{x})}$$
$$= \frac{\sum_{c' \in \mathcal{U}} z_{c'}(\boldsymbol{x})}{\sum_{c' \in \mathcal{S}} z_{c'}(\boldsymbol{x}) + \sum_{c' \in \mathcal{U}} z_{c'}(\boldsymbol{x})} \times \frac{z_c(\boldsymbol{x})}{\sum_{c' \in \mathcal{U}} z_{c'}(\boldsymbol{x})},$$

SoftMax decomposition: probability that ***x*** belongs to an **absent** class c

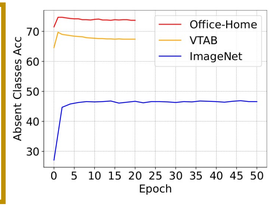
probability that ***x*** belongs to any **absent** classes *U*

probability that within the absent classes U, ***x*** belongs to class c.



FT model tends to classify **absent** class as **fine-tuning** classes

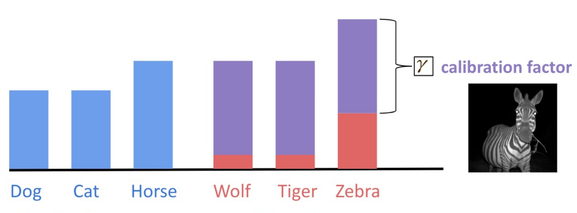
FT model's ability to discriminate absent class sample **within absent classes** is **improved**

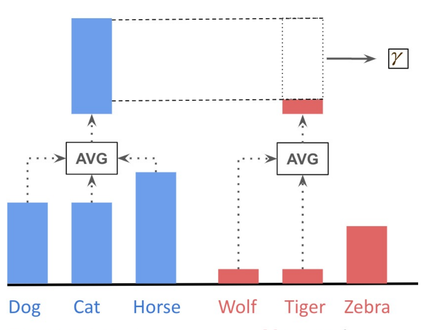


Root cause: FT model's **biased logits** towards fine-tuning classes

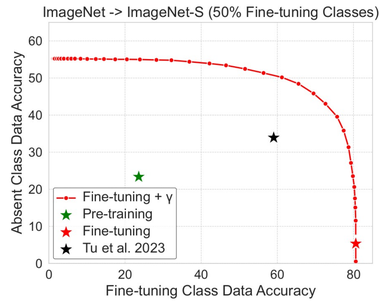
Post-Processing Calibration for the Rescue

$$\hat{y} = \arg\max_{c \in \mathcal{Y}} \quad \boldsymbol{w}_c^\top f_\theta(\boldsymbol{x}) + \gamma \mathbb{1}[c \in \mathcal{U}].$$



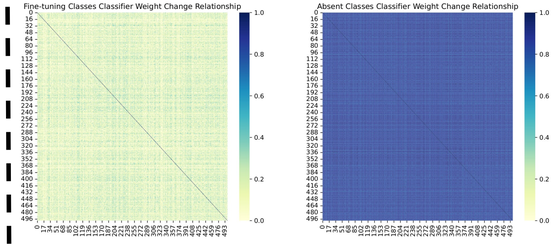


Results



- **FT (*) + post processing calibration** outperforms SOTA (*)
- Observations are **robust** to data split, fine-tuning class size and optimizer.

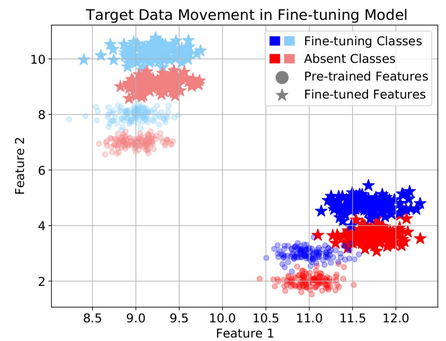
Why is absent class relationship preserved?



Similarity of classifier weight changes from pre-trained model to FT model for each class

Compared with **fine-tuning** classes, **the weight changes for absent classes are highly similar**

Why do absent class features improve after FT?



Toy example:

- **absent** class features moving in a **similar direction** as **nearby fine-tuning** class features.

Acknowledgement: This research is supported by grants from the NSF (ICICLE: OAC2112606).