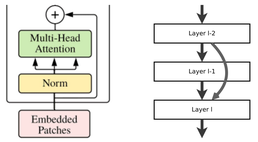
# ModelDiff: A Framework for Comparing Learning Algorithms

Harshay Shah*, Sung Min Park*, Andrew Ilyas*, Aleksander Mądry

## Comparing Learning Algorithms

### ML pipelines entail many design choices

Model architecture

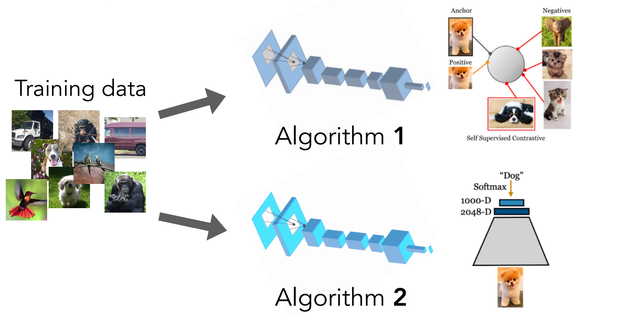



Transformers **or** ResNets?

Augmentation schemes

Random Crop **or** Flip **or** Median Blur?

### Recurring Q: Which pipeline to choose?

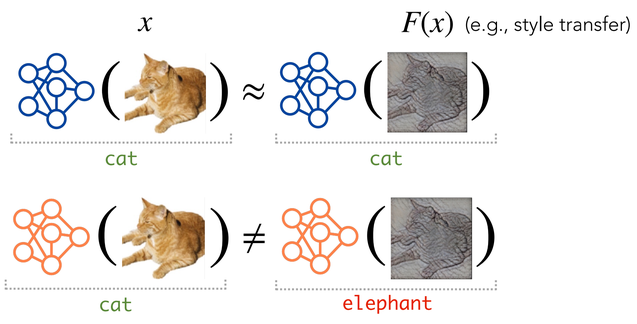



**Conventional approach:** Compare model performance

## Algorithm Comparisons with ModelDiff

**Problem:** *Identify differences between algorithm 1 and algorithm 2 in a fine-grained way*

**How?** *Find input-space* **distinguishing transformation** $F$ *with disparate impact on algorithm 1 and algorithm 2*




## ModelDiff in three steps

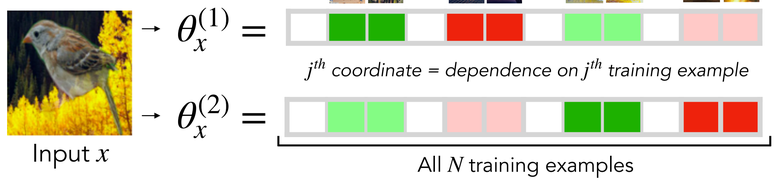
**Case study:** Compare models trained on Waterbirds data

**Alg 1:** Fine-tune ImageNet model **Alg 2:** Train from scratch

### Step 1: Compute datamodels for both algorithms

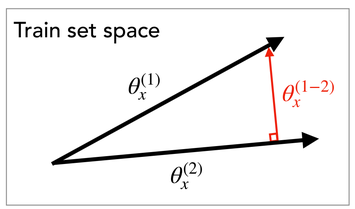
[IPE+22]

Datamodel $\theta_x$ identifies training examples that impact prediction on $x$




### Step 2: Find distinguishing subpopulations

**Key idea:** Use datamodels to compare how training examples influence models trained with algorithm 1 and algorithm 2

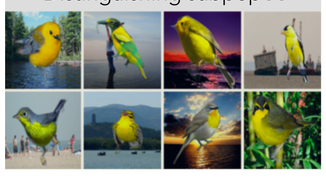



Datamodels $\theta_x^{(1)}$ (alg 1) and $\theta_x^{(2)}$ (alg 2) share the same train set space!

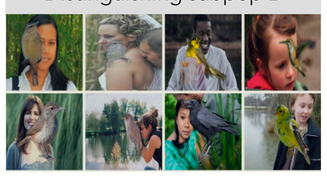
Residual datamodel $\theta_x^{(1-2)}$ identifies training examples important for alg 1 but not alg 2

**Distinguishing subpopulations**: Clusters of test inputs on which algorithm 1 and algorithm 2 rely on different training examples

Distinguishing subpop **A**



Distinguishing subpop **B**



**Approach:** Use PCA to cluster residual datamodels

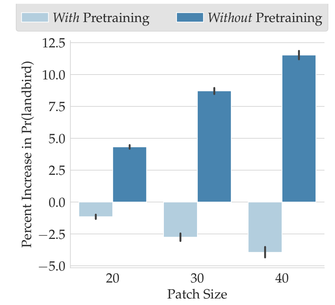
### Step 3: Infer + test distinguishing transformations

Inspect extracted subpopulations to **infer** distinguishing transformation and **test** its effect on both alg 1 and alg 2

No ImageNet pre-training → "yellow color" bias

Subpop **A** surfaces "yellow color" subpop


"Yellow color" transformation

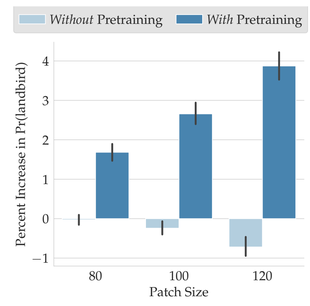



$\Pr(\hat{y} = \text{landbird} \mid \text{do(yellow color)})$

ImageNet pre-training → "human face" bias

Subpop **B** surfaces "human face" subpop


"Human face" transformation




$\Pr(\hat{y} = \text{landbird} \mid \text{do(human face)})$

### Takeaways

→ **ModelDiff**: Fine-grained comparisons of learning algorithms

→ **Use-case**: Pinpoint train-time design choices shape model biases

→ **Main idea**: Compare impact of training examples on predictions

**Paper** **Code** **Blog post**