Exploring and Utilizing Pattern Imbalance

Shibin Mei, Chenglong Zhao, Shengchao Yuan, Bingbing Ni*
Shanghai Jiao Tong University, Shanghai 200240, China
{adair327, cl-zhao, sc_yuan, nibingbing}@sjtu.edu.cn

CVPR 2023

OOD Generalization

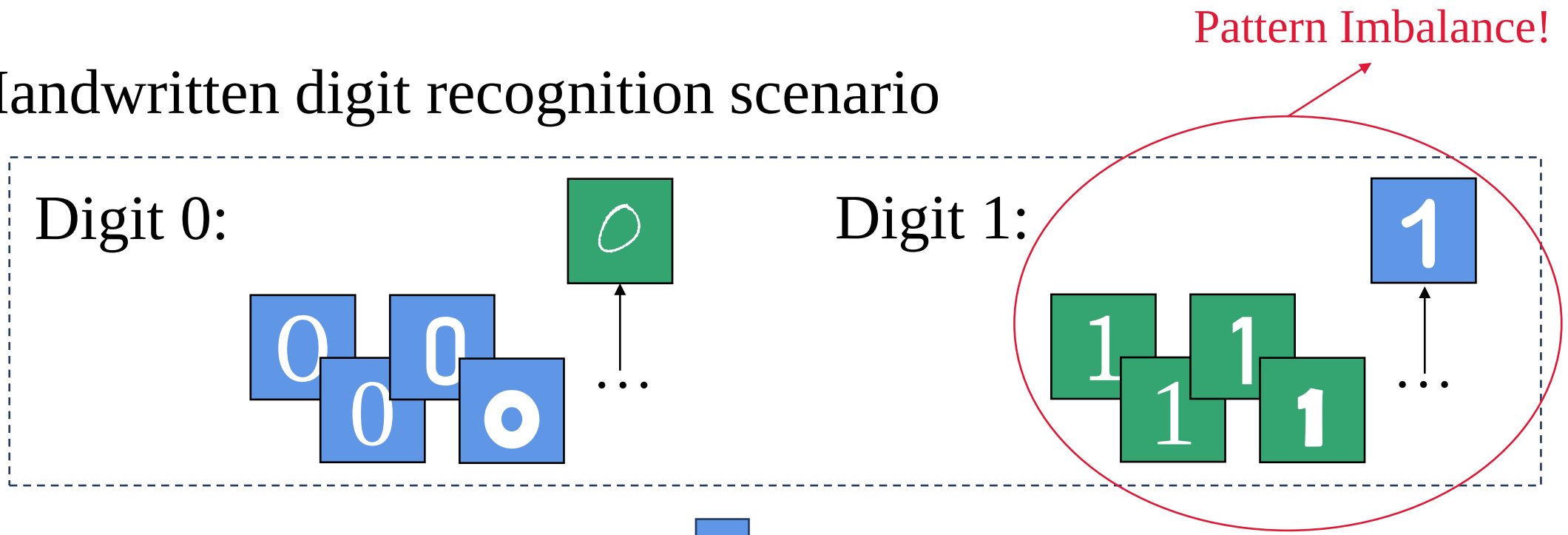
- Some researchers focus on encouraging the model to learn domain invariant features.

- Other researchers start by avoiding spurious features.

Most of methods manually design specific model structures to handle domain generalization.

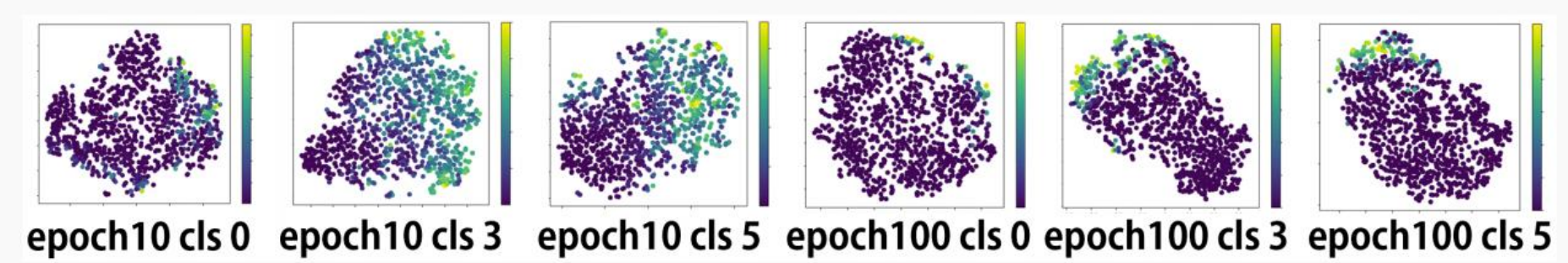
Instead of designing specific networks, we are more concerned about exploring the character of the dataset.

Handwritten digit recognition scenario



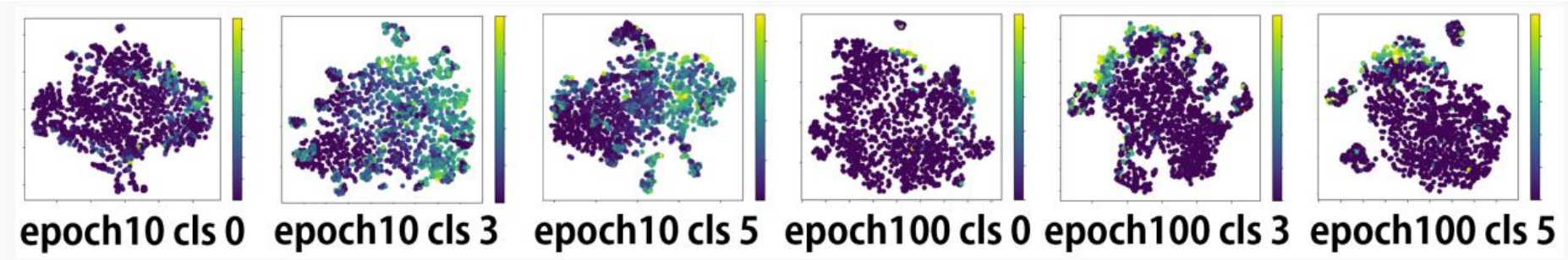
Is it classifying backgrounds or classifying digit?

From Data View



(a) Representation space

From Model View



(b) Activation path

From Optimization View

Different patterns have different training extent. The classification loss is used to characterize the extent of optimization.

The alignment of these views

Pair	Random	Rep-Path	Rep-Loss	Path-Loss
CCT	0.118	0.241	0.577	0.508

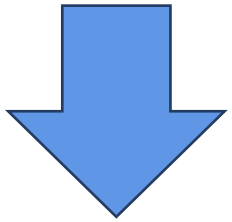
$$CCT_{AB} = \mathbb{E}_{g \in \mathcal{G}} \mathbb{E}_{s_i, s_j \in g, R^A_{s_i} = R^A_{s_j}} [\mathbb{1}(R^B_{s_i} = R^B_{s_j})]$$

Pattern imbalance exists!

Analysis

We attribute the OOD generalization problem to the mining of hard or minority patterns under imbalanced patterns.

we argue that the model should not learn the knowledge reflected by a single difficult sample, but the knowledge contained in a group of special samples with similar characteristics, that is, more macro patterns.



Seed Category

Definition 1 *(Seed) For a given dataset $\mathcal{D}$ and the above concept of adjacent samples, we define seed set $\mathcal{S}$ if and only if for any sample $x \in \mathcal{D} - \mathcal{S}$, there exists at least one $s \in \mathcal{S}$ that satisfies x and s are adjacent samples. We define each sample in the smallest seed set $\mathcal{S}$ as seed.*

Definition 2 *(Seed Category) For any seed s, we define the union of this seed and the samples $\mathcal{B}_s$ subordinate to it as seed category, i.e.,*

$$\mathcal{C}_{seed} = \{s\} \cup \mathcal{B}_s. \tag{4}$$

Algorithm

Algorithm 1 Dynamic Distribution Based on Seed Category

Require: Original data $\mathcal{D}$, total training steps T, re-distribution cycle t_0

Ensure: Model parameters θ

1: **for** t= 1 to T **do**
2: Conduct re-distribution and update seed category for $\mathcal{D}$ every t_0 steps.
3: Maintain a weight distribution vector q on these seed categories.
4: Sample data from each seed category as a batch.
5: Calculate average losses and update weights for each seed category.
6: Weight normalization.
7: Update model parameters θ according to distribution weight vector q.
8: **end for**

Experiment

Algorithm	ERM	DANN	IRM	GDRO	MLDG	MMD	MTL	ARM	SagNet	VREx	Ours
Train-domain validation set											
A	81.1	81.9	82.7	86.2	81.8	84.9	82.5	82.1	83.0	81.4	82.0
C	81.6	77.8	78.5	80.5	80.0	81.0	79.9	82.9	78.6	81.4	80.3
P	97.0	95.1	96.4	96.2	95.3	95.7	95.5	93.6	95.3	96.0	96.8
S	74.1	75.4	74.3	75.3	69.5	73.3	79.6	76.0	80.7	77.8	80.8
Avg.	83.5	82.6	83.0	84.5	81.6	83.7	84.4	83.6	84.4	84.2	**85.0**
Leave-one-domain-out cross validation											
A	82.7	79.0	82.7	81.3	82.6	81.6	80.0	77.2	80.7	81.8	83.5
C	80.0	76.1	78.5	76.8	79.5	80.8	80.3	82.9	78.1	79.9	79.9
P	95.3	95.1	96.4	94.8	97.7	95.1	96.7	93.1	95.7	95.4	96.2
S	75.3	72.4	74.3	80.3	70.5	71.7	74.6	73.2	63.6	72.8	83.2
Avg.	83.3	78.6	83.0	83.3	82.5	82.3	82.9	81.6	79.6	82.4	**85.7**
Test-domain validation set											
A	80.9	74.0	72.4	72.4	82.6	81.2	84.5	73.5	77.5	81.8	80.2
C	81.2	75.6	77.1	79.0	81.3	81.7	76.1	76.7	78.6	79.7	79.6
P	95.1	91.0	90.9	94.8	94.9	95.1	94.2	94.7	95.7	95.3	94.4
S	78.1	76.1	74.1	72.6	73.2	78.8	74.6	70.6	77.4	76.3	84.3
Avg.	83.8	79.2	78.9	79.7	83.0	84.2	82.3	78.9	82.3	83.3	**84.6**

PACS dataset

Experiment

Algorithm	ERM	DANN	IRM	GDRO	MLDG	MMD	MTL	ARM	SagNet	VREx	Ours
Train-domain validation set											
0.1	71.6	71.5	61.0	72.5	72.4	49.3	71.8	72.8	71.4	72.3	72.8
0.2	72.6	73.2	67.8	72.2	72.6	63.4	71.6	72.6	73.5	73.0	73.2
0.9	9.8	10.0	9.8	10.2	10.1	10.8	10.2	10.1	10.3	10.2	10.7
Avg.	51.3	51.6	46.2	51.6	51.7	41.2	51.2	51.8	51.7	51.8	**52.2**
Leave-one-domain-out cross validation											
0.1	61.4	50.9	48.8	50.9	49.2	49.5	47.8	47.4	50.7	72.2	55.6
0.2	50.5	50.1	50.2	51.0	53.3	50.6	66.8	50.6	49.4	50.7	71.8
0.9	9.8	10.0	9.8	10.2	10.1	9.8	10.2	10.1	10.3	10.0	10.2
Avg.	40.6	37.0	40.6	37.3	37.5	36.7	41.6	36.0	36.8	44.3	**45.9**
Test-domain validation set											
0.1	64.6	69.7	51.0	67.1	70.1	50.5	67.2	77.8	64.6	72.2	71.8
0.2	68.5	70.6	62.3	68.4	71.0	50.6	68.8	70.3	68.7	71.7	72.5
0.9	26.5	13.5	50.6	38.5	23.8	10.3	20.1	17.5	28.4	29.4	37.1
Avg.	53.2	51.3	54.7	58.0	54.9	37.1	52.1	55.2	53.9	57.8	**60.5**

Table 3. Evaluation of the domain generalization ability of the proposed method compared with recent OOD algorithms on the ColoredMNIST dataset. 0.1, 0.2, 0.9 represent the color flip probability of three domains. All values are shown in percentages.

Experiment

Datasets	ColoredMNIST						PACS					
Methods	Loss			Cluster			Loss			Cluster		
Seeds per domain	2	3	4	2	3	4	2	3	4	2	3	4
Model Selection 1	51.5	51.5	**52.2**	51.6	51.6	51.9	82.9	83.0	**85.0**	82.5	82.6	82.6
Model Selection 2	38.5	41.8	**45.9**	37.3	37.3	37.5	82.8	84.2	**85.7**	83.5	83.5	83.8
Model Selection 3	59.2	57.4	**60.5**	54.8	59.3	53.0	83.1	79.6	**84.6**	82.2	81.8	80.9

Table 4. Performance vs. seed category calculation methods.

Experiment

Datasets	CMNIST		RMNIST		PACS		OfficeHome		VLCS	
Methods	Baseline	Ours	Baseline	Ours	Baseline	Ours	Baseline	Ours	Baseline	Ours
PCMA(G)	0.186	0.186	0.437	0.437	8.112	8.124	8.113	8.128	8.117	8.125
TTR(min)	6.18	7.14	7.75	9.62	44.15	48.14	66.53	71.19	76.57	81.71

Table 5. Efficiency of our method compared with baseline. PCMA represents peak CUDA memory allocated and TTR represents the time of a training round.

Experiment

Algo.	A	C	P	R	Avg.
ERM	54.7	47.3	72.7	74.0	62.2
DANN	54.3	51.1	73.0	67.4	61.5
IRM	55.4	49.1	68.2	75.0	61.9
GDRO	55.7	52.0	71.4	74.7	63.4
MTL	53.0	47.1	70.5	76.3	61.7
ARM	51.9	46.8	69.8	71.0	59.9
SagNet	53.1	49.0	72.5	73.4	62.0
Ours	57.5	50.4	73.2	74.0	**63.8**

Table 6. Evaluation on OfficeHome datasets. All values are shown in percentages. A, C, P, R represent four domains of OfficeHome, i.e., Art, Clipart, Product, Real World.

Algo.	C	L	S	V	Avg.
ERM	98.1	59.0	70.1	74.6	75.4
DANN	98.5	63.0	56.9	74.6	73.2
IRM	95.4	59.3	74.2	76.0	76.2
GDRO	94.9	66.3	69.7	71.3	75.6
MTL	96.1	59.5	70.0	73.0	74.6
ARM	94.6	62.9	74.0	70.3	75.4
SagNet	93.4	60.4	75.1	75.0	76.0
Ours	97.4	66.4	70.1	72.8	**76.7**

Table 7. Evaluation on VLCS datasets. All values are shown in percentages. C, L, S, V represent four domains of VLCS, i.e., Caltech101, LabelMe, SUN09, VOC2007.

Thanks