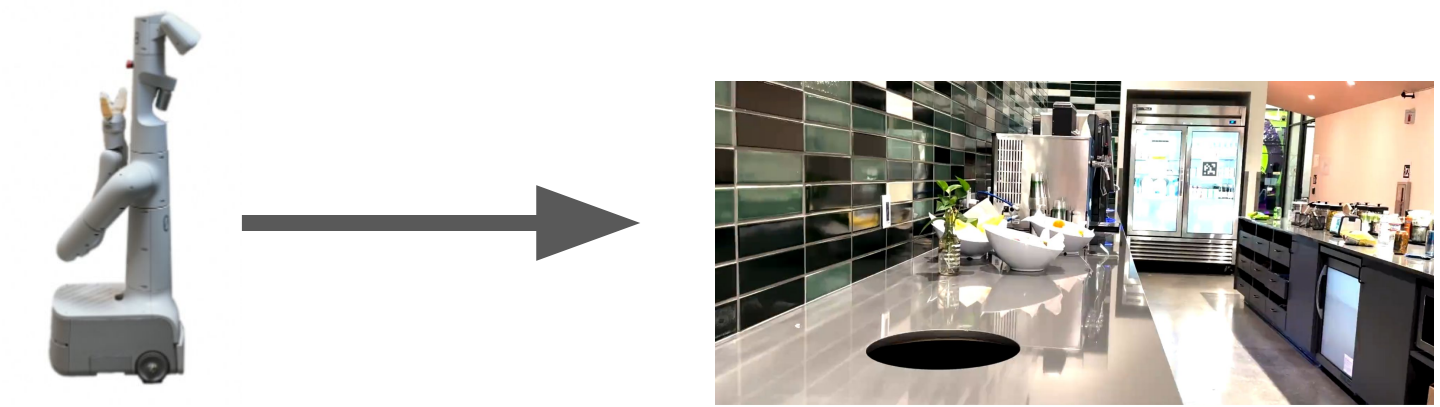
Efficient Test-Time Adaptation

Related works and Project Status

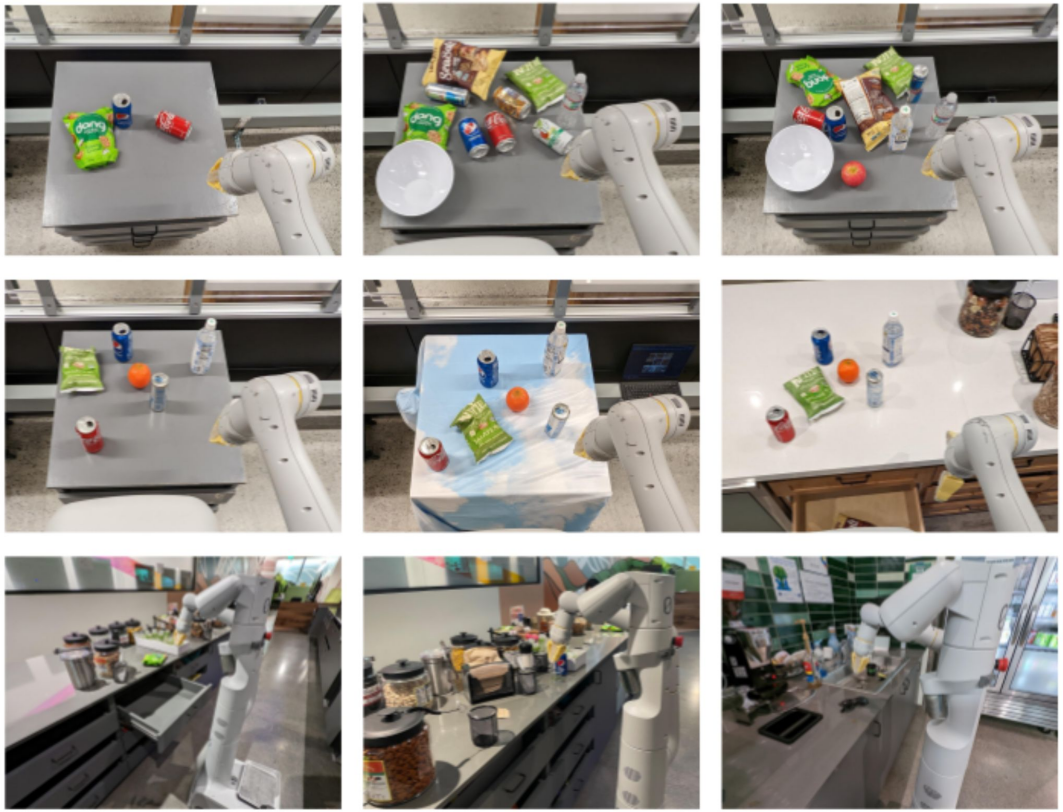
Juan Camacho, Andrea De Carlo, Samuele Bolotta

Motivation



RT-1: Robotics Transformer for Real-World Control at Scale, Google Research, 2022

Motivation



Robustness to the number of distractor objects

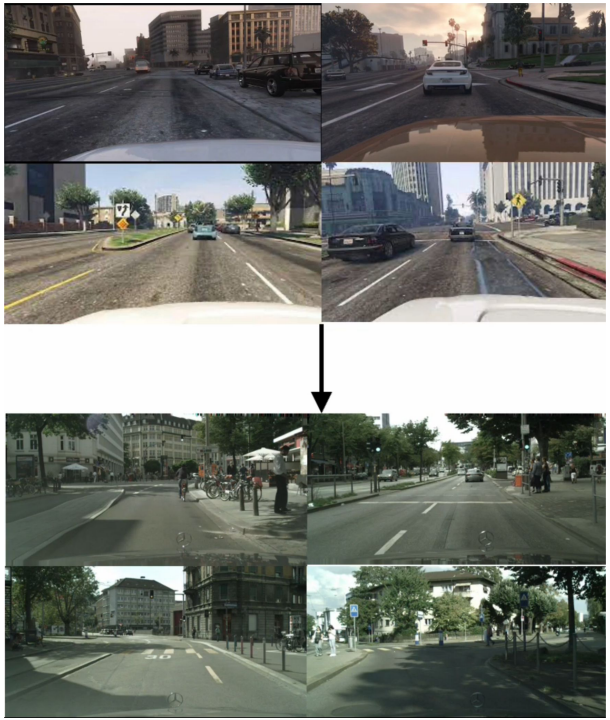
robustness to different backgrounds and environments such as new, previously unseen kitchens

realistic scenarios that combine all these elements

RT-1: Robotics Transformer for Real-World Control at Scale, Google Research, 2022

Motivation

Sim to real

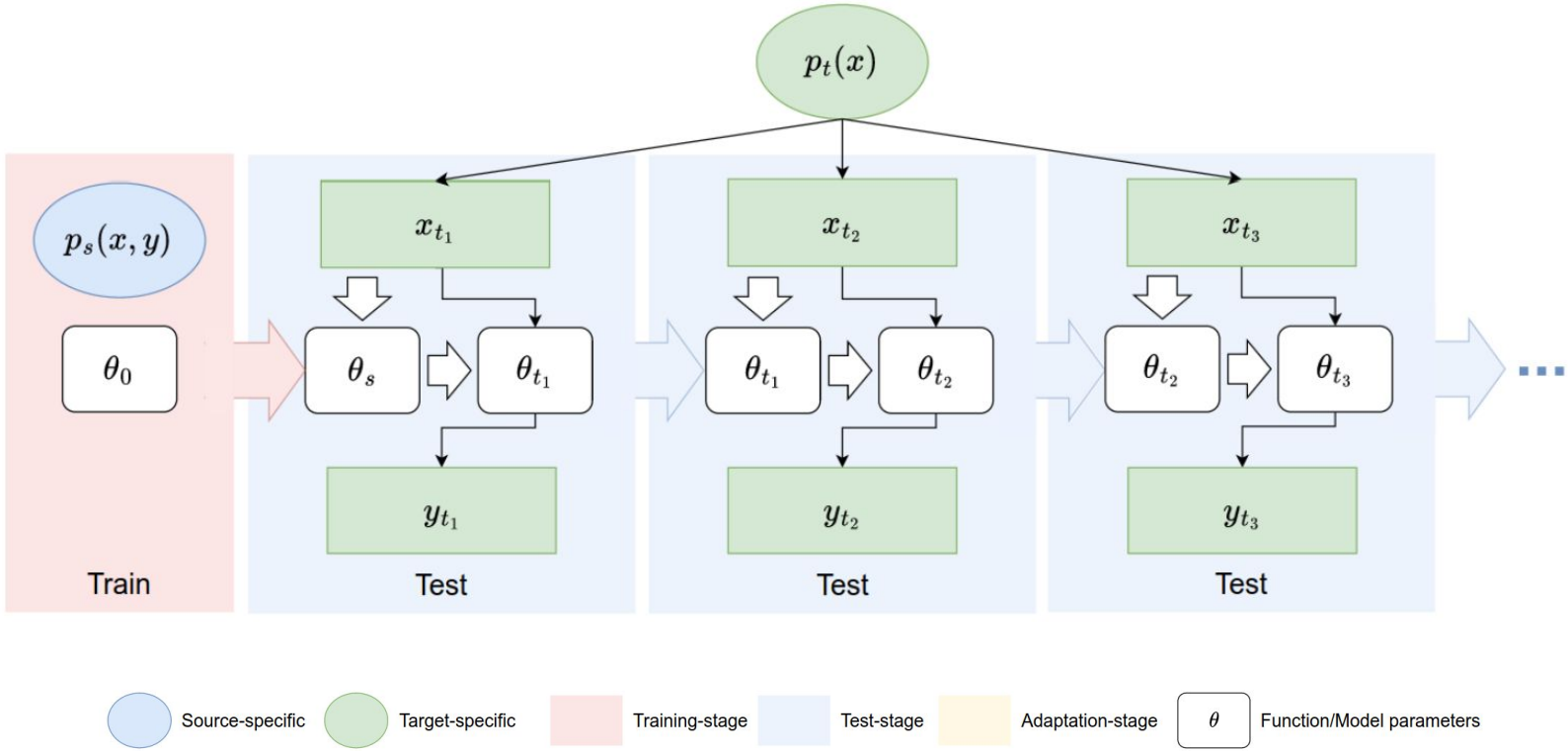


Different conditions

Motivation


CHRIS SIMONS
CAUGHT ON CAMERA
WAYMO STRUGGLES TO STAY IN ITS LANE
INSTAGRAM VIDEO SHOWS SELF-DRIVING CAR SWERVING BACK AND FORTH ON 15TH AVE.
6:18
89°
STOCKS
NASDAQ: 16,794.88
108.91
GIFRUN.COM

But what is Test-Time Adaptation (TTA)?



Xiao e Snoek "Beyond Model Adaptation at Test Time: A Survey"

TTA addresses distribution shifts

$$p(x_t, y_t) \neq p(x_s, y_s)$$

$$p(x, y) = p(x)p(y|x) = p(y)p(x|y)$$

Covariate shifts

$$p(x_t) \neq p(x_s)$$
$$p(y_t|x_t) = p(y_s|x_s)$$

Label shifts

$$p(y_t) \neq p(y_s)$$
$$p(x_t|y_t) = p(x_s|y_s)$$

Concept shifts

$$p(x_t) \neq p(x_s)$$
$$p(y_t|x_t) = p(y_s|x_s)$$

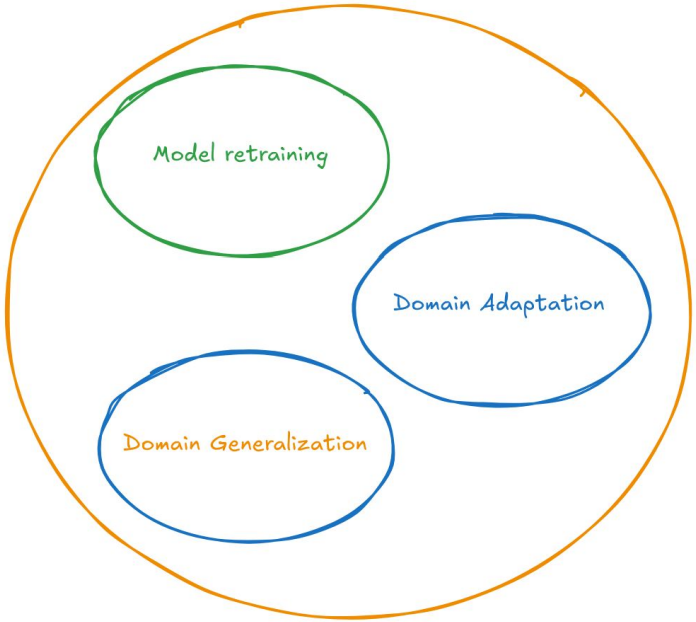
Conditional shifts

$$p(y_t) = p(y_s)$$
$$p(x_t|y_t) \neq p(x_s|y_s)$$

Xiao e Snoek “Beyond Model Adaptation at Test Time: A Survey”

Why Test-Time Adaptation?

Traditional solutions



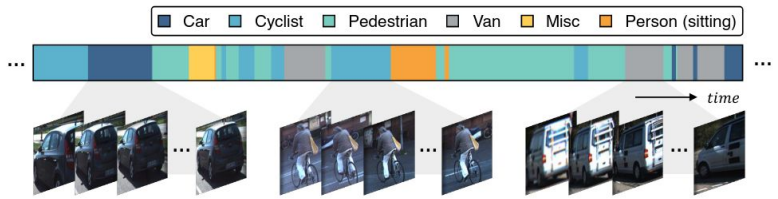
TTA advantages

- Online Adaptation
- Resource efficiency
- Practical benefits

Xiao e Snoek “Beyond Model Adaptation at Test Time: A Survey”

Understanding data streams

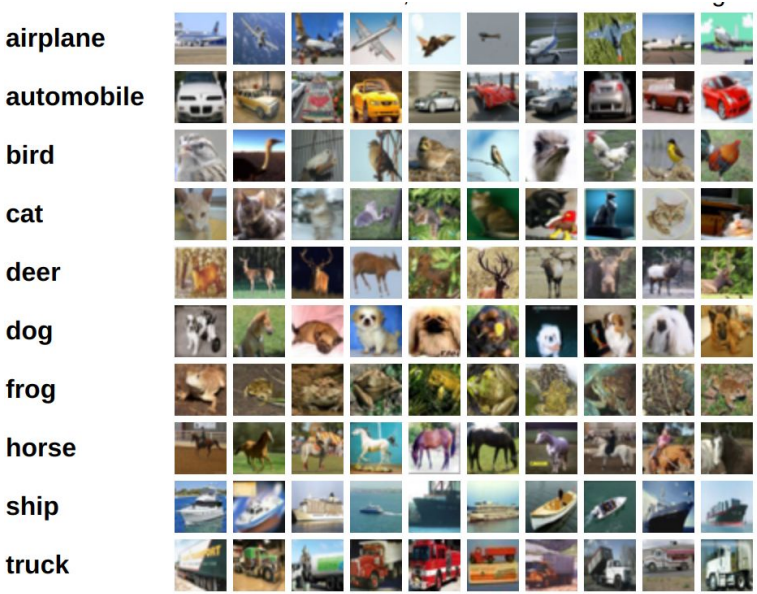
Non-I.I.D. Data Stream



(a) KITTI dataset.

(b) HARTH dataset.

I.I.D. Data Stream



CIFAR-10

NOTE: Robust Continual Test-time Adaptation Against Temporal Correlation

Understanding data streams

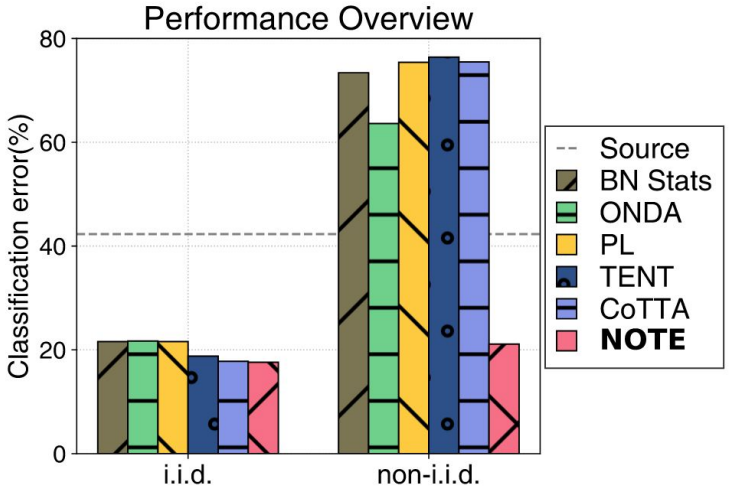


Figure 2: Average classification error (%) of existing TTA methods and our method (NOTE) on CIFAR10-C [13]. The error rates significantly increase under the non-i.i.d. setting compared with the i.i.d. setting. Lower is better.

Impact of Data Streams on TTA

I.I.D. Data Streams

- Simpler adaptation methods
- Efficiency
- Fast performance

Non-I.I.D. Data Streams

- Complex adaptation strategies
- Temporal aware
- Resource heavy

Key Trade-offs

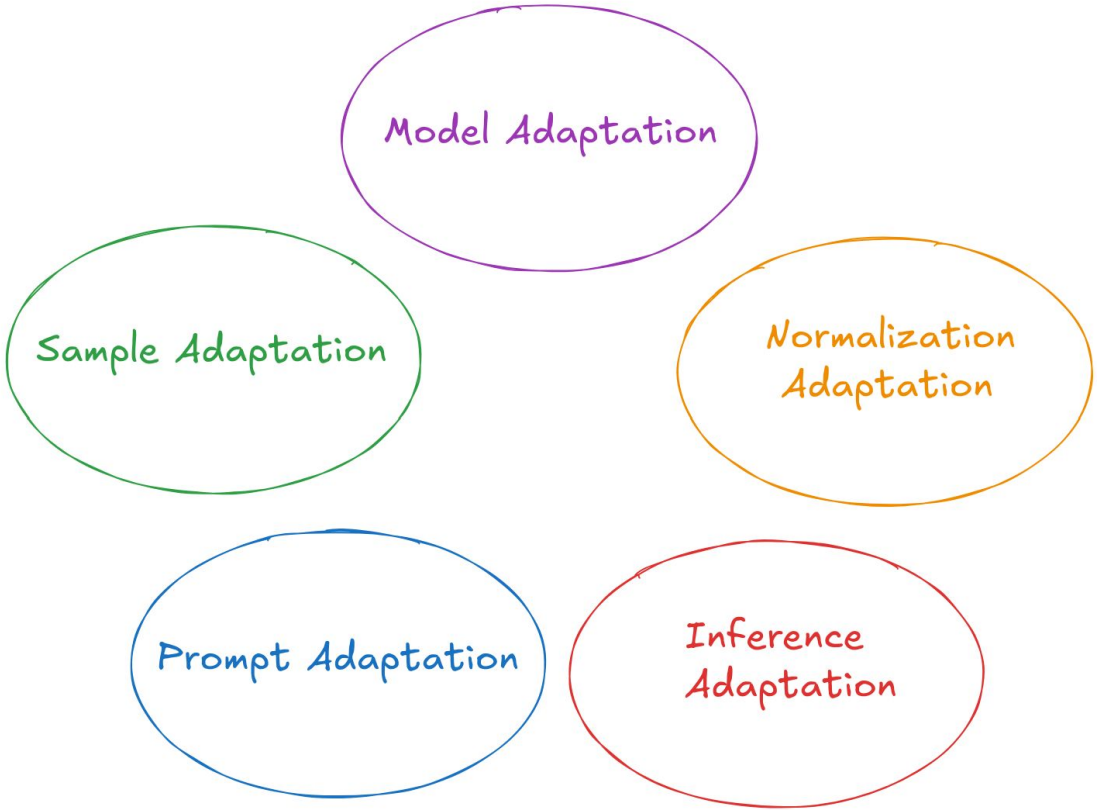
🎯 Accuracy vs Speed

🧠 Stability vs Plasticity

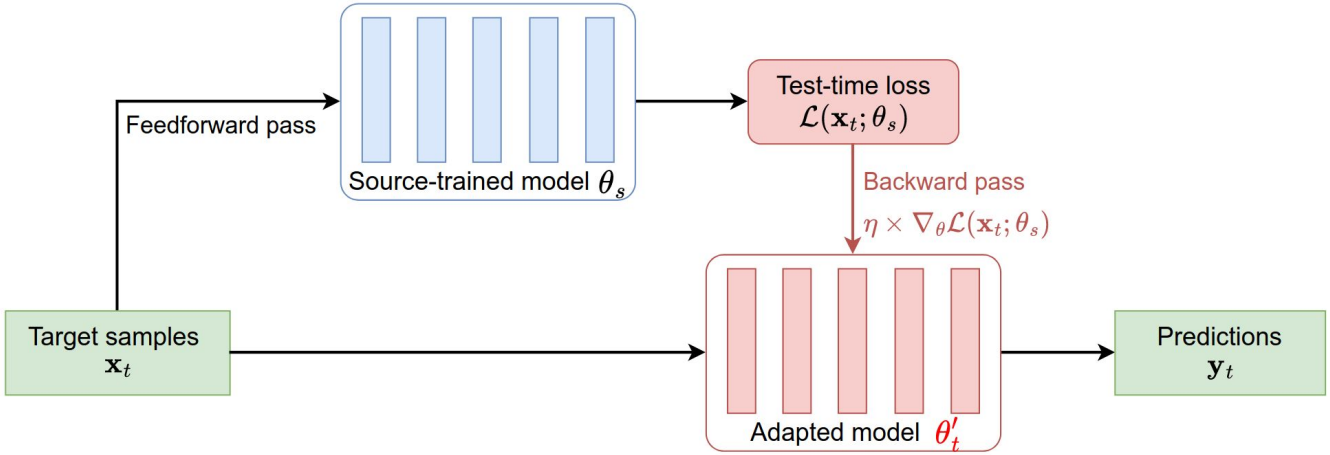
💻 Complexity vs Efficiency

NOTE: Robust Continual Test-time Adaptation Against Temporal Correlation

Test-Time Adaptation Methods

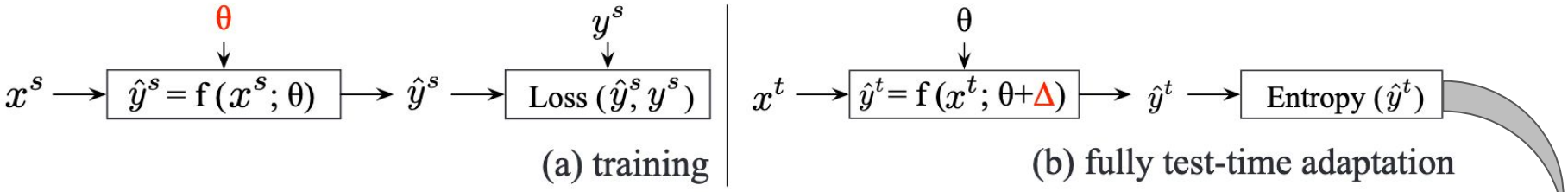


Model Adaptation

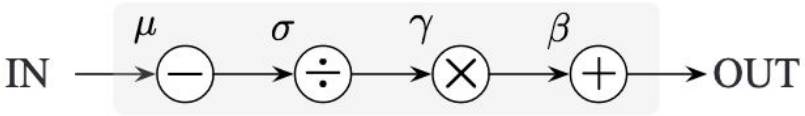


Xiao e Snoek “Beyond Model Adaptation at Test Time: A Survey”

Model Adaptation



$$H(\hat{y}) = -\sum_c p(\hat{y}_c) \log p(\hat{y}_c)$$

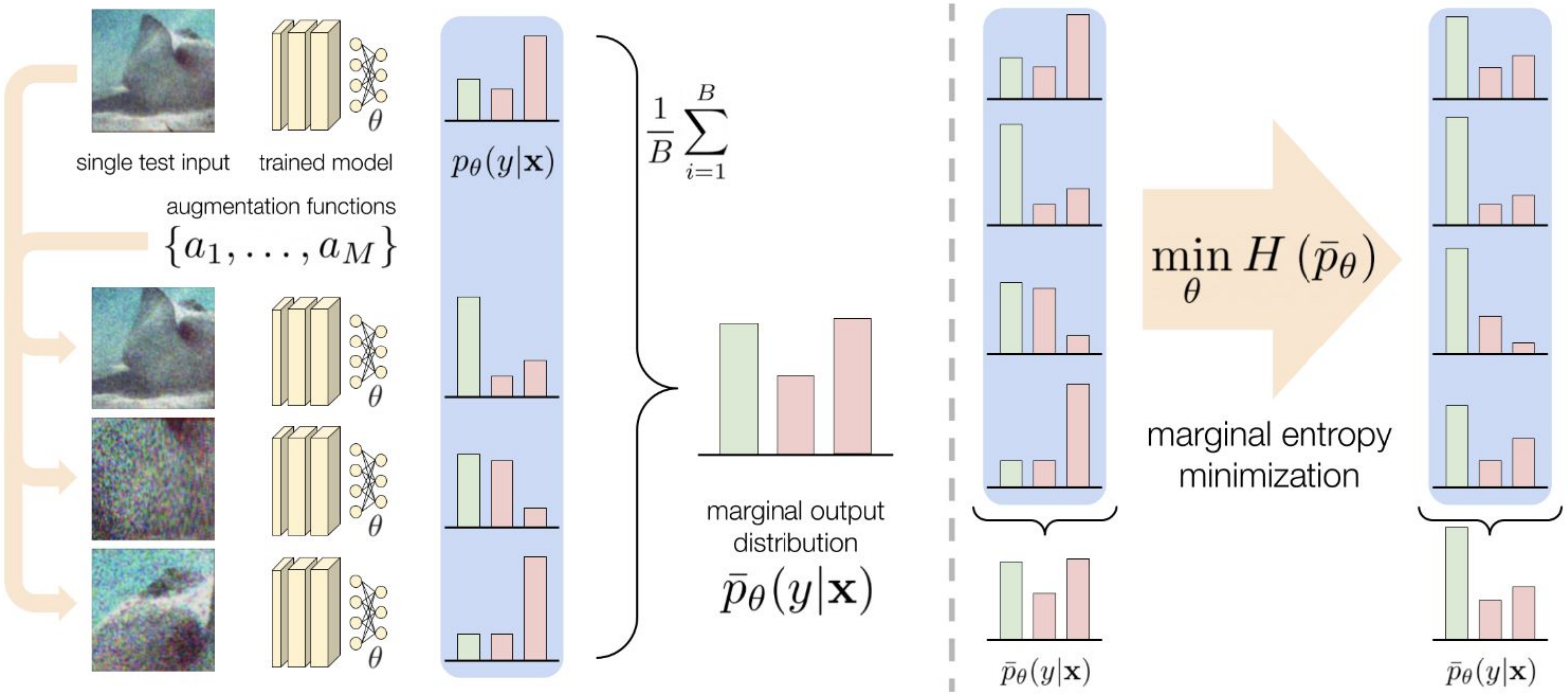


normalization $\mu \leftarrow \mathbb{E}[x_t], \sigma^2 \leftarrow \mathbb{E}[(\mu - x_t)^2]$

transformation $\gamma \leftarrow \gamma + \partial H/\partial\gamma, \beta \leftarrow \beta + \partial H/\partial\beta$

Wang et al. “Tent: Fully Test-time Adaptation by Entropy Minimization”

Model Adaptation



Zhang et al. “MEMO: Test Time Robustness via Adaptation and Augmentation”

Model Adaptation

Test-time adaptation with “zero” temperature

Marginal probability distribution:

$$\overline{p}(\cdot|\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} p(\cdot|\mathbf{x}_i)$$

Marginal entropy loss:

$$\mathcal{L}_{ent} = H(\overline{p}(\cdot|\mathbf{x})) = -\sum_{c=1}^{C} \overline{p}(y=c|\mathbf{x}) \log(\overline{p}(y=c|\mathbf{x}))$$

Farina et al. “Frustratingly Easy Test-Time Adaptation of Vision-Language Models”

Model Adaptation

Test-time adaptation with “zero” temperature

Marginal entropy loss:

$$\mathcal{L}_{ent} = H(\overline{p}(\cdot|\mathbf{x})) = -\sum_{c=1}^{C} \overline{p}(y=c|\mathbf{x}) \log(\overline{p}(y=c|\mathbf{x}))$$

Re-written loss in TPT:

$$\mathcal{L}_{ent} = H(\overline{p}(\cdot|\mathbf{x}, \mathbf{t}_{\text{ctx}})) = -\sum_{c}^{C} \overline{p}(y=c|\mathbf{x}, \mathbf{t}_{\text{ctx}}, \tau) \log\ (\overline{p}(y=c|\mathbf{x}, \mathbf{t}_{\text{ctx}}, \tau))$$

$$\text{where } \overline{p}(y=c|\mathbf{x}, \mathbf{t}_{\text{ctx}}, \tau) = \frac{1}{N} \sum_{i}^{N} \frac{\exp\left(\mathbf{z}_i^{\text{img}} \cdot \mathbf{z}_c^{\text{txt}}(\mathbf{t}_{\text{ctx}})/\tau\right)}{\sum_k^C \exp\left(\mathbf{z}_i^{\text{img}} \cdot \mathbf{z}_k^{\text{txt}}(\mathbf{t}_{\text{ctx}})/\tau\right)}.$$

Farina et al. “Frustratingly Easy Test-Time Adaptation of Vision-Language Models”

Model Adaptation

Test-time adaptation with zero temperature

```
def zero(image, z_txt, N, gamma, temp):
  """
  :param z_txt: pre-computed text embeddings (C,hdim)
  :param temp: model's original temperature
  :param augment: takes (C,H,W) and returns (N,C,H,W)
  :param gamma: filtering percentile (e.g., 0.3)
  """
  views = augment(image, num_views=N) # generate augmented views
  l = model.image_encoder(views) @ z_txt.t() # predict (unscaled logits)
  l_filt = confidence_filter(l, temp, top=gamma) # retain most confident preds
  zero_temp = torch.finfo(l_filt.dtype).eps # zero temperature
  p_bar = (l_filt / zero_temp).softmax(dim=1).sum(dim=0) # marginalize
  return p_bar.argmax()
```

Farina et al. “Frustratingly Easy Test-Time Adaptation of Vision-Language Models”

Model Adaptation

Test-time adaptation with zero temperature

Table 2: Fine-grained classification. TTA methods are grouped according to the reference baseline, top-1 accuracy is reported and **bold text** indicates the best performer of each group.

Method	FLWR	DTD	PETS	CARS	UCF	CAL	FOOD	SUN	AIR	ESAT	Mean	Median
						CLIP-ViT-B-16						
Zero-Shot	67.44	44.27	88.25	65.48	65.13	93.35	83.65	62.59	23.67	42.01	63.58	65.31
Ensemble	67.07	45.09	**88.28**	66.16	67.51	93.91	84.04	66.26	23.22	**50.42**	65.20	66.66
TPT	**68.75**	**47.04**	87.23	66.68	68.16	93.93	84.67	65.39	23.13	42.86	64.78	67.42
ZERO	67.68	46.12	87.75	68.04	67.77	93.66	86.53	65.03	**25.21**	34.33	64.21	67.72
$\text{ZERO}_{\text{+Ensemble}}$	67.17	45.86	87.83	**68.97**	**69.18**	**94.41**	**86.77**	**67.63**	**25.21**	42.17	**65.52**	**68.30**
						MaPLe						
Zero-Shot	72.23	46.49	90.49	65.57	68.69	93.53	86.20	67.01	24.74	**48.06**	66.30	67.85
TPT	72.37	45.87	90.72	66.50	69.19	93.59	86.64	67.54	24.70	47.80	66.49	68.36
PromptAlign	**72.39**	47.24	**90.76**	68.50	69.47	94.01	86.65	67.54	24.80	47.86	**66.92**	68.99
ZERO	71.62	**47.89**	90.60	**68.58**	**69.87**	**94.48**	**87.20**	**68.20**	**26.25**	39.47	66.42	**69.23**
						CLIP-ViT-B-16 + CLIP-ViT-L-14						
Zero-Shot	75.76	51.83	92.86	76.16	73.70	94.04	88.03	66.96	30.54	**54.38**	70.43	74.73
RLCF $^{\mathbf{t}_{\text{ctx}}}_{t=1}$	71.58	50.34	89.01	69.76	69.84	94.09	85.90	67.33	23.71	46.87	66.84	69.80
RLCF $^{\mathbf{t}_{\text{ctx}}}_{t=3}$	72.49	51.93	89.55	72.91	72.31	95.00	86.84	69.04	25.40	45.96	68.14	72.40
RLCF $^{\Theta_v}_{t=1}$	72.56	52.21	89.51	63.12	71.49	94.65	86.90	68.50	24.06	47.74	67.07	70.00
RLCF $^{\Theta_v}_{t=3}$	71.74	53.27	91.15	70.93	73.24	94.73	87.28	69.38	28.54	47.41	68.77	71.34
ZERO	**76.41**	**53.63**	**94.08**	**78.39**	**74.68**	**95.21**	**90.66**	**69.61**	**33.62**	44.21	**71.05**	**75.55**

Farina et al. “Frustratingly Easy Test-Time Adaptation of Vision-Language Models”

Model Adaptation

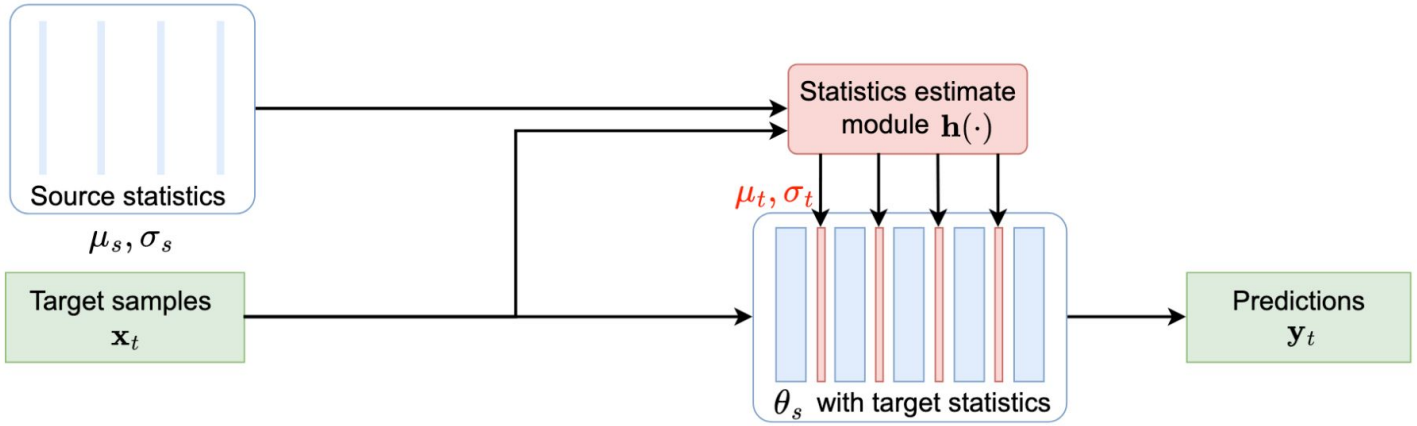
Test-time adaptation with zero temperature

Table 3: Computational requirements of different TTA methods.

Metric	CLIP-ViT-B-16		CLIP-ViT-B-16 + CLIP-ViT-L-14		
	TPT	ZERO	RLCF $^{\mathbf{t}_{\text{ctx}}}_{t=3}$	RLCF $^{\Theta_v}_{t=3}$	ZERO
Time [s]	0.57±0.01	**0.06**±0.01	1.20±0.02	0.18±0.01	**0.08**±0.02
Mem [GB]	17.66	**1.40**	18.64	9.04	**2.58**

Farina et al. “Frustratingly Easy Test-Time Adaptation of Vision-Language Models”

Normalization Adaptation



$$\hat{\mathbf{x}}_s = \frac{\mathbf{x}_s - \mu_s}{\sqrt{\sigma_s^2 + \epsilon}} \longrightarrow \mu_t, \sigma_t = \mathbf{h}(\mathbf{x}_t, \mu_s, \sigma_s), \quad \hat{\mathbf{x}}_t = \frac{\mathbf{x}_t - \mu_t}{\sqrt{\sigma_t^2 + \epsilon}}, \quad \mathbf{y}_t = f_{\boldsymbol{\theta}_s}(\hat{\mathbf{x}}_t),$$

Xiao e Snoek “Beyond Model Adaptation at Test Time: A Survey”

Normalization Adaptation

$$\mu_t, \sigma_t = \mathbf{h}(\mathbf{x}_t, \mu_s, \sigma_s), \quad \hat{\mathbf{x}}_t = \frac{\mathbf{x}_t - \mu_t}{\sqrt{\sigma_t^2 + \epsilon}}, \quad \mathbf{y}_t = f_{\boldsymbol{\theta}_s}(\hat{\mathbf{x}}_t),$$

Target statistics

- Directly estimate
- Alleviate covariate shift

Statistics combination

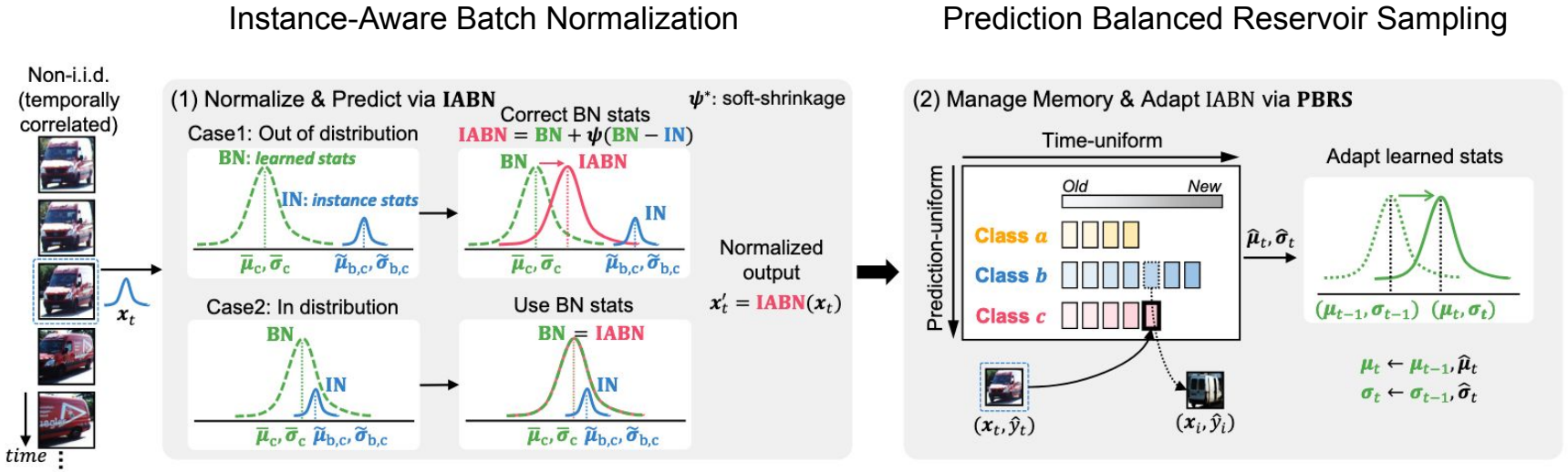
- Weighed sum
- adaptation ⚖️ preservation

Statistics inference

- Meta Learning per sample
- No requirements on data

Xiao e Snoek “Beyond Model Adaptation at Test Time: A Survey”

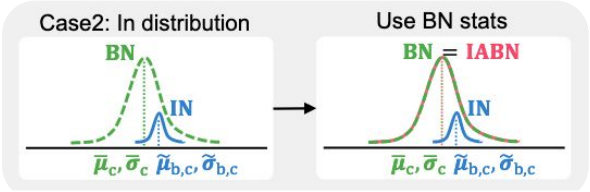
Normalization Adaptation

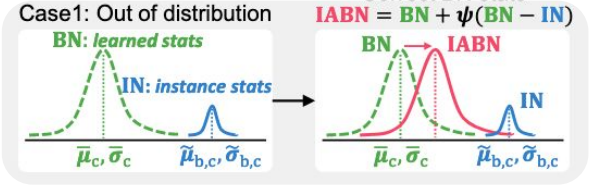


T. Gong et al. “NOTE: Robust Continual Test-time Adaptation Against Temporal Correlation”

Normalization Adaptation

Instance-Aware Batch Normalization





$$\mathrm{BN}(\mathbf{f}_{:,c,:}; \boldsymbol{\mu}_c, \boldsymbol{\sigma}_c^2) := \gamma \cdot \frac{\mathbf{f}_{:,c,:} - \boldsymbol{\mu}_c}{\sqrt{\boldsymbol{\sigma}_c^2 + \epsilon}} + \beta$$

$$\mathrm{IABN}(\mathbf{f}_{b,c,:}; \bar{\boldsymbol{\mu}}_c, \bar{\boldsymbol{\sigma}}_c^2; \tilde{\boldsymbol{\mu}}_{b,c}, \tilde{\boldsymbol{\sigma}}_{b,c}^2) := \gamma \cdot \frac{\mathbf{f}_{b,c,:} - \boldsymbol{\mu}_{b,c}^{\mathrm{IABN}}}{\sqrt{(\boldsymbol{\sigma}_{b,c}^{\mathrm{IABN}})^2 + \epsilon}} + \beta$$

$$\hat{\boldsymbol{\mu}}_c := \frac{1}{BL} \sum_{b,l} \mathbf{f}_{b,c,l}$$

$$\hat{\boldsymbol{\sigma}}_c^2 := \frac{1}{BL} \sum_{b,l} (\mathbf{f}_{b,c,l} - \hat{\boldsymbol{\mu}}_c)^2$$

$$\boldsymbol{\mu}_{b,c}^{\mathrm{IABN}} := \bar{\boldsymbol{\mu}}_c + \psi(\tilde{\boldsymbol{\mu}}_{b,c} - \bar{\boldsymbol{\mu}}_c; \alpha s_{\tilde{\boldsymbol{\mu}},c})$$

$$(\boldsymbol{\sigma}_{b,c}^{\mathrm{IABN}})^2 := \bar{\boldsymbol{\sigma}}_c^2 + \psi(\tilde{\boldsymbol{\sigma}}_{b,c}^2 - \bar{\boldsymbol{\sigma}}_c^2; \alpha s_{\tilde{\boldsymbol{\sigma}}^2,c})$$

$$s_{\tilde{\boldsymbol{\mu}},c}^2 := \frac{\bar{\boldsymbol{\sigma}}_c^2}{L}$$

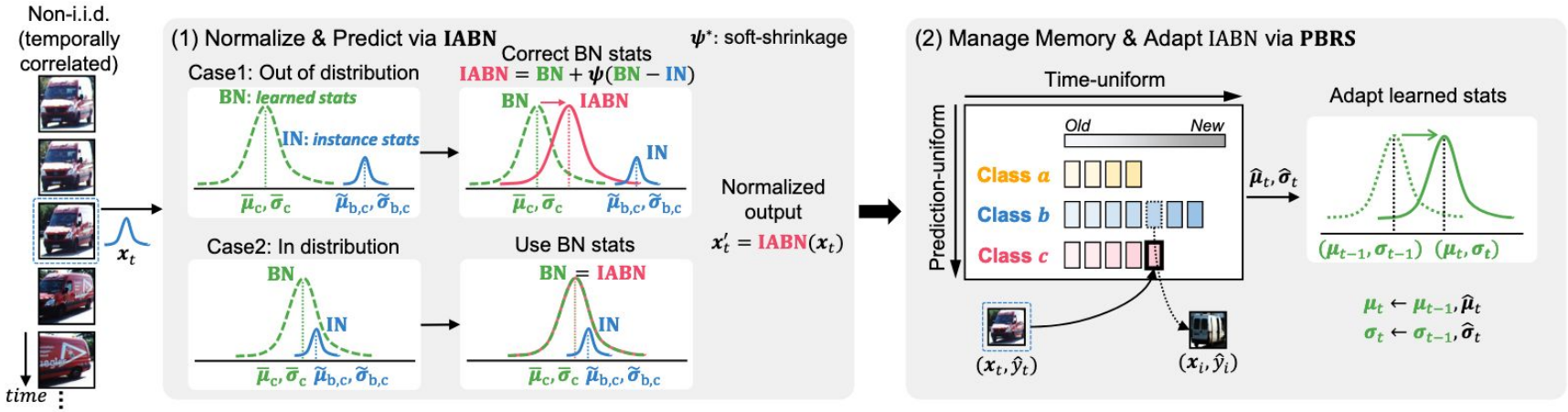
$$s_{\tilde{\boldsymbol{\sigma}}^2,c}^2 := \frac{2\bar{\boldsymbol{\sigma}}_c^4}{L-1}$$

T. Gong et al. “NOTE: Robust Continual Test-time Adaptation Against Temporal Correlation”

Normalization Adaptation

Instance-Aware Batch Normalization

Prediction Balanced Reservoir Sampling



$$\mathrm{IABN}(\mathbf{f}_{b,c,:}; \bar{\boldsymbol{\mu}}_c, \bar{\boldsymbol{\sigma}}_c^2; \tilde{\boldsymbol{\mu}}_{b,c}, \tilde{\boldsymbol{\sigma}}_{b,c}^2) := \gamma \cdot \frac{\mathbf{f}_{b,c,:} - \boldsymbol{\mu}_{b,c}^{\mathrm{IABN}}}{\sqrt{(\boldsymbol{\sigma}_{b,c}^{\mathrm{IABN}})^2 + \epsilon}} + \beta$$

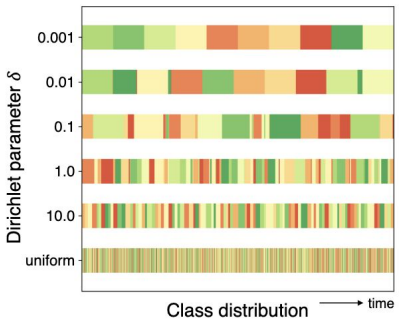
$$\boldsymbol{\mu}_t = (1-m)\boldsymbol{\mu}_{t-1} + m\frac{N}{N-1}\hat{\boldsymbol{\mu}}_t$$

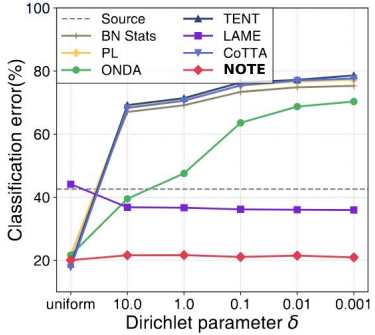
$$\boldsymbol{\sigma}_t^2 = (1-m)\boldsymbol{\sigma}_{t-1}^2 + m\frac{N}{N-1}\hat{\boldsymbol{\sigma}}_t^2$$

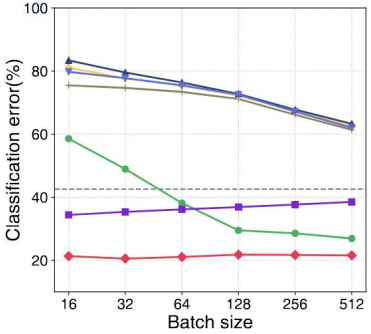
T. Gong et al. “NOTE: Robust Continual Test-time Adaptation Against Temporal Correlation”

Normalization Adaptation

Method	Temporally correlated test stream				Uniformly distributed test stream			
	CIFAR10-C	CIFAR100-C	ImageNet-C	Avg	CIFAR10-C	CIFAR100-C	ImageNet-C	Avg
Source	42.3 ± 1.1	66.6 ± 0.1	86.1 ± 0.0	65.0	42.3 ± 1.1	66.6 ± 0.1	86.1 ± 0.0	65.0
BN Stats [29]	73.4 ± 1.3	65.0 ± 0.3	96.9 ± 0.0	78.5	21.6 ± 0.4	46.6 ± 0.2	76.0 ± 0.0	48.1
ONDA [27]	63.6 ± 1.0	49.6 ± 0.3	89.0 ± 0.0	67.4	21.7 ± 0.4	46.5 ± 0.1	75.9 ± 0.0	48.0
PL [22]	75.4 ± 1.8	66.4 ± 0.4	98.9 ± 0.0	80.2	21.6 ± 0.2	43.1 ± 0.3	74.4 ± 0.2	46.4
TENT [41]	76.4 ± 2.7	66.9 ± 0.6	96.9 ± 0.0	80.1	18.8 ± 0.2	**40.3 ± 0.2**	76.0 ± 0.0	45.0
LAME [4]	36.2 ± 1.3	63.3 ± 0.3	82.7 ± 0.0	60.7	44.1 ± 0.5	68.8 ± 0.1	86.3 ± 0.0	66.4
CoTTA [44]	75.5 ± 0.7	64.2 ± 0.2	97.0 ± 0.0	78.9	17.8 ± 0.3	44.3 ± 0.2	71.5 ± 0.0	44.6
NOTE	**21.1 ± 0.6**	**47.0 ± 0.1**	**80.6 ± 0.1**	**49.6**	20.1 ± 0.5 (**17.6 ± 0.3**)	46.4 ± 0.0 (41.0 ± 0.2)	**70.3 ± 0.0** (71.7 ± 0.0)	45.6 (**43.4**)







T. Gong et al. “NOTE: Robust Continual Test-time Adaptation Against Temporal Correlation”

Normalization Adaptation

$$\mu_t, \sigma_t = \mathbf{h}(\mathbf{x}_t, \mu_s, \sigma_s), \quad \hat{\mathbf{x}}_t = \frac{\mathbf{x}_t - \mu_t}{\sqrt{\sigma_t^2 + \epsilon}}, \quad \mathbf{y}_t = f_{\boldsymbol{\theta}_s}(\hat{\mathbf{x}}_t),$$

Target statistics

- Directly estimate
- Alleviate covariate shift

Large distribution shifts ➔ mismatch

Statistics combination

- Weighed sum
- adaptation ⚖️ preservation

Hyperparameters

Statistics inference

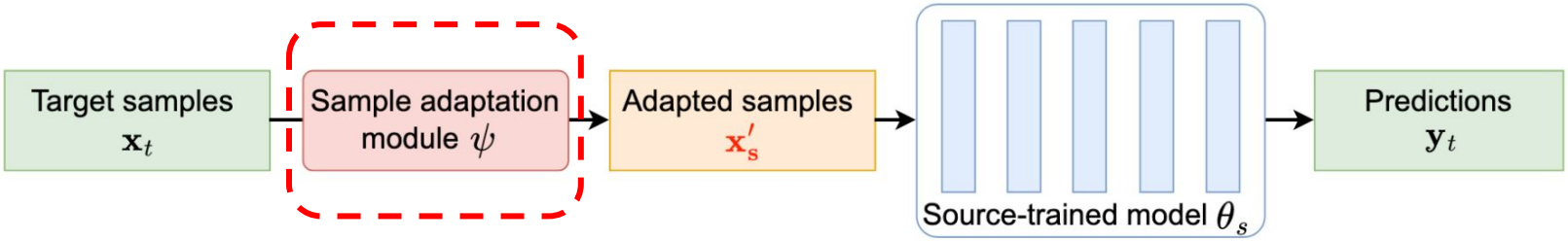
- Meta Learning per sample
- No requirements on data

Difficult to learn

NOT model-agnostic

Xiao e Snoek “Beyond Model Adaptation at Test Time: A Survey”

Sample Adaptation



$$\mathbf{x}'_s = \psi(\mathbf{x}_t), \quad \mathbf{y}_t = f_{\theta_s}(\mathbf{x}'_s).$$

Xiao e Snoek “Beyond Model Adaptation at Test Time: A Survey”

Sample Adaptation

$$\mathbf{x}'_s = \psi(\mathbf{x}_t), \quad \mathbf{y}_t = f_{\boldsymbol{\theta}_s}(\mathbf{x}'_s).$$

Feature adjustment

- Learned to extract useful information
- Efficients and easy In feature space

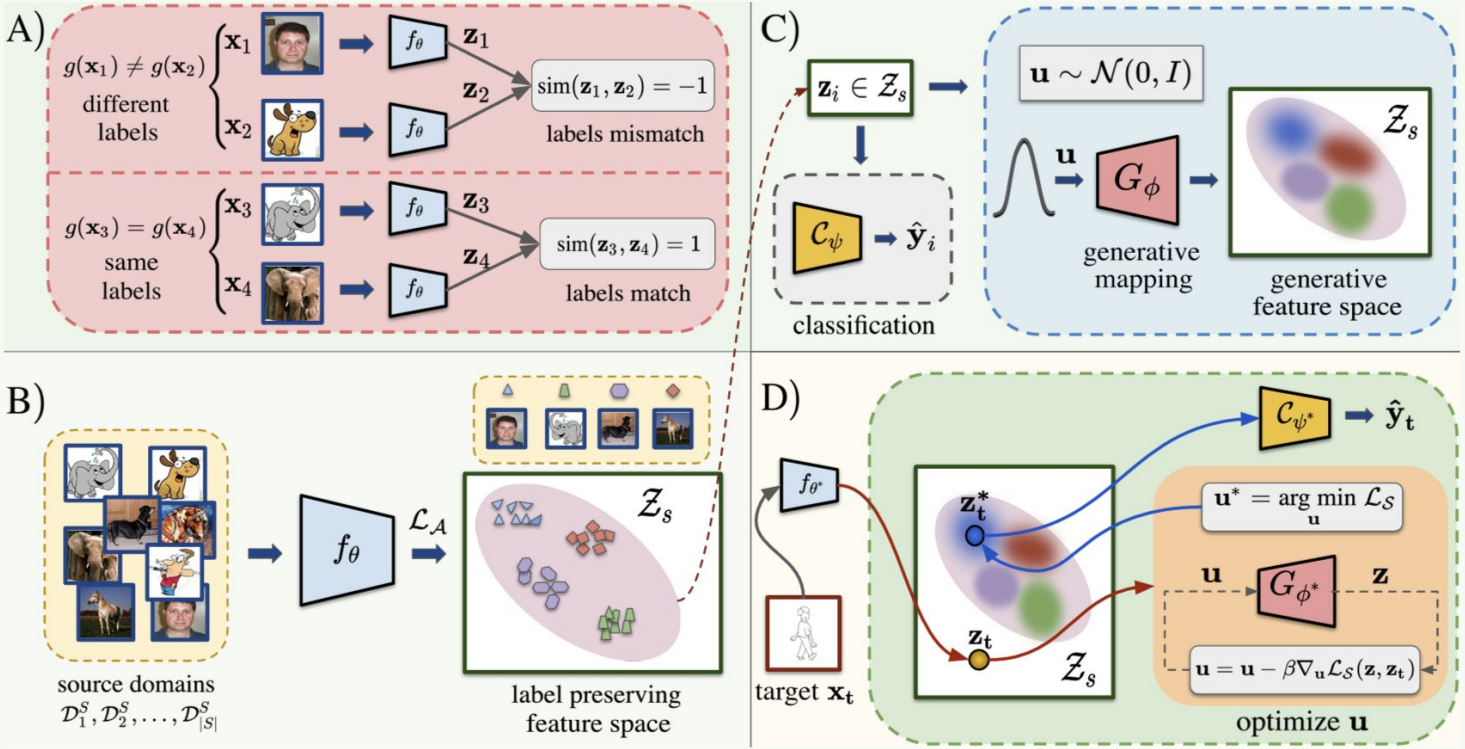
Generative models

Input adjustment

- Advances in generative models

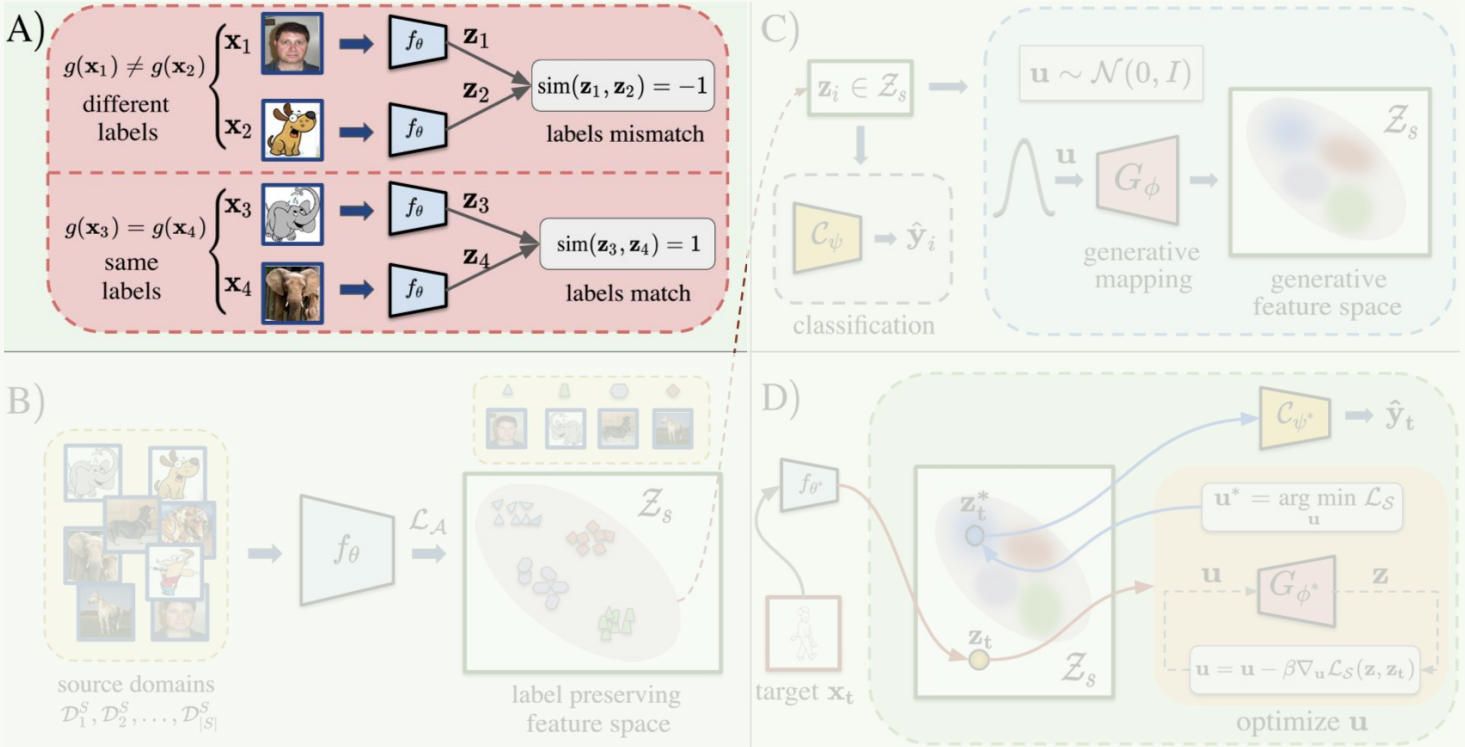
Xiao e Snoek “Beyond Model Adaptation at Test Time: A Survey”

Sample Adaptation



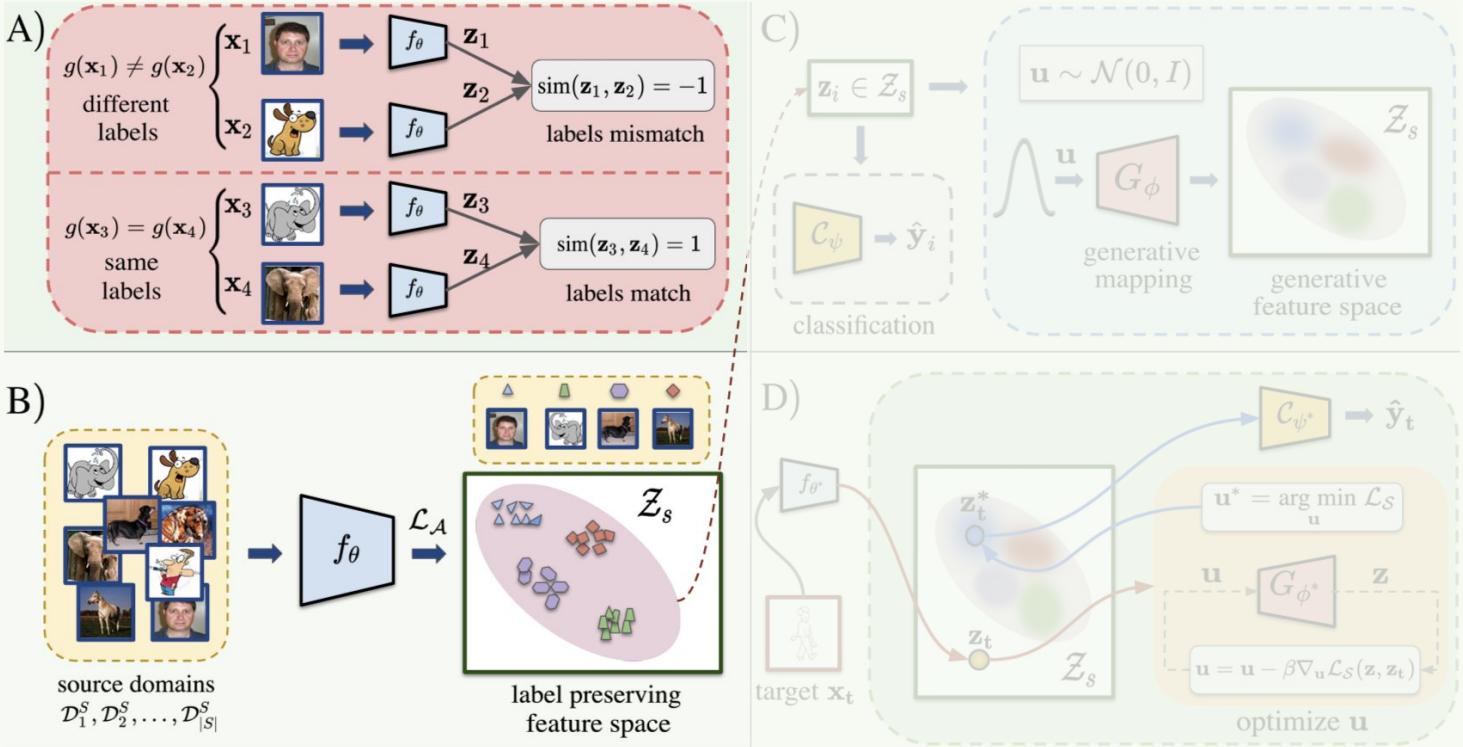
Pandey et al. “Generalization on Unseen Domains via Inference-time Label-Preserving Target Projections”

Sample Adaptation



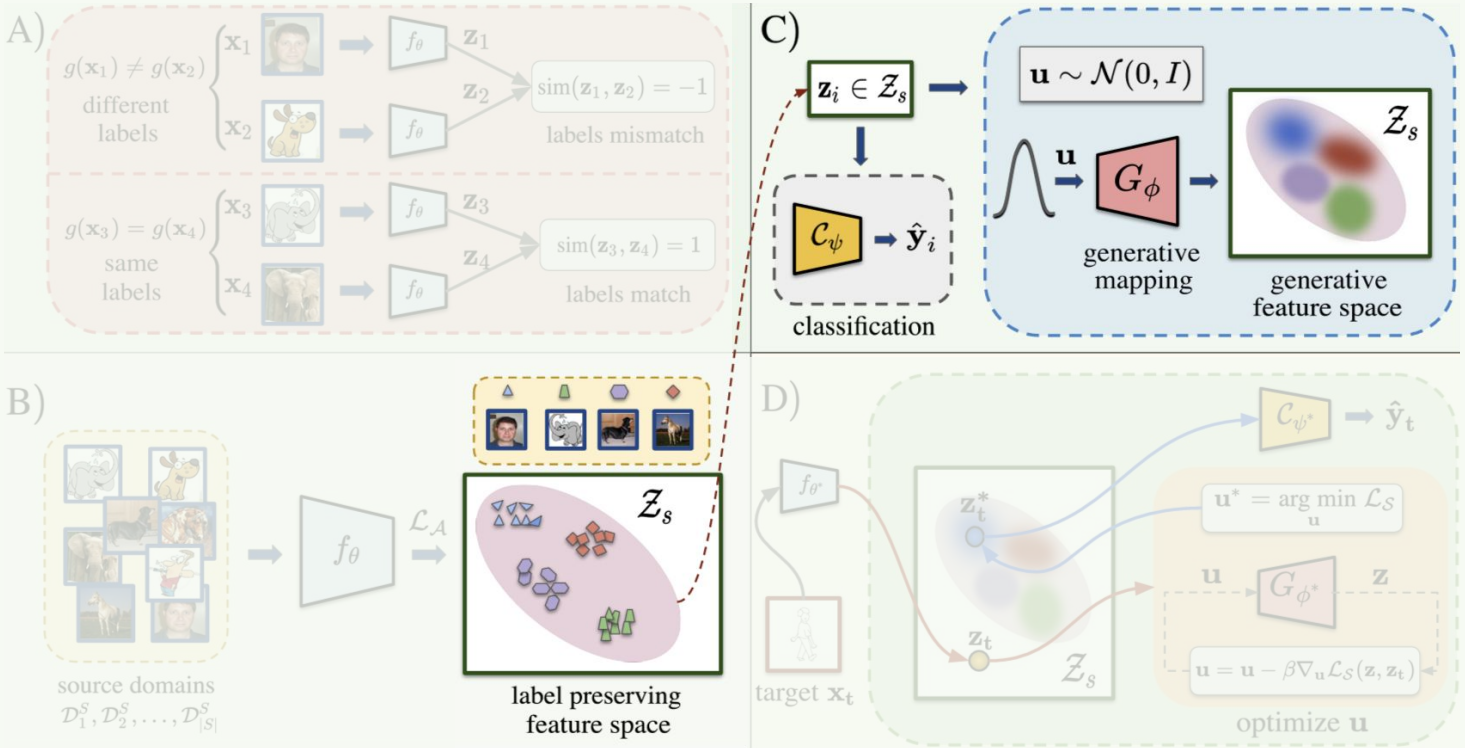
Pandey et al. “Generalization on Unseen Domains via Inference-time Label-Preserving Target Projections”

Sample Adaptation



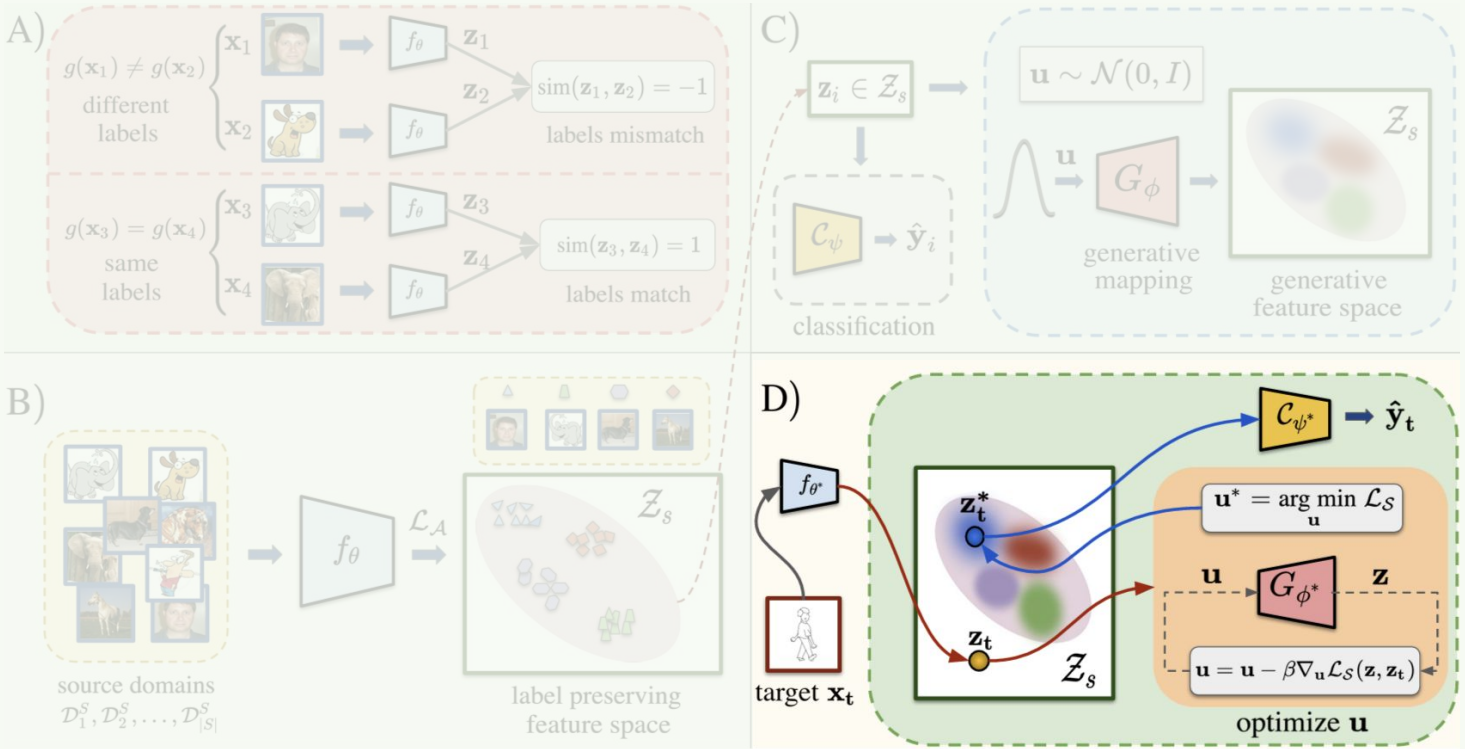
Pandey et al. “Generalization on Unseen Domains via Inference-time Label-Preserving Target Projections”

Sample Adaptation



Pandey et al. “Generalization on Unseen Domains via Inference-time Label-Preserving Target Projections”

Sample Adaptation



Pandey et al. “Generalization on Unseen Domains via Inference-time Label-Preserving Target Projections”

Sample Adaptation

Method	MNIST	MNIST-M	SVHN	SYN	Avg.
Deep All	95.24±0.1	58.36±0.6	62.12±0.5	78.94±0.3	73.66
Jigen [5]	96.50±0.0	61.40±0.0	63.70±0.0	74.00±0.0	73.90
CCSA [29]	95.20±0.0	58.20±0.0	65.50±0.0	79.10±0.0	74.50
MMD-AAE [24]	96.50±0.0	58.40±0.0	65.00±0.0	78.40±0.0	74.60
CrossGrad [38]	96.70±0.0	61.10±0.0	65.30±0.0	80.20±0.0	75.80
L2A-OT [47]	96.70±0.0	63.90±0.0	68.60±0.0	83.20±0.0	78.10
Ours	**97.99±0.1**	**66.52±0.4**	**71.31±0.3**	**85.40±0.5**	**80.30**

Method	Artistic	Clipart	Product	Real-World	Avg.
Deep All	52.06±0.5	46.12±0.3	70.45±0.2	72.45±0.2	60.27
D-SAM [7]	58.03±0.0	44.37±0.0	69.22±0.0	71.45±0.0	60.77
Jigen [5]	53.04±0.0	47.51±0.0	71.47±0.0	72.79±0.0	61.20
MMD-AAE [24]	56.50±0.0	47.30±0.0	72.10±0.0	74.80±0.0	62.70
DSON [37]	59.37±0.0	45.70±0.0	71.84±0.0	74.68±0.0	62.90
RSC [16]	58.42±0.0	47.90±0.0	71.63±0.0	74.54±0.0	63.12
L2A-OT [47]	60.60±0.0	50.10±0.0	74.80±0.0	77.00±0.0	65.60
Ours	**62.63±0.2**	**55.79±0.3**	**76.86±0.1**	**78.98±0.1**	**68.56**

Method	Caltech	LabelMe	Pascal	Sun	Avg.
Deep All	96.45±0.1	60.03±0.5	70.41±0.4	62.63±0.3	72.38
Jigen [5]	96.93±0.0	60.90±0.0	70.62±0.0	64.30±0.0	73.19
MMLD [27]	96.66±0.0	58.77±0.0	71.96±0.0	68.13±0.0	73.88
MASF [6]	94.78±0.2	64.90±0.1	69.14±0.2	67.64±0.1	74.11
EISNet [45]	97.33±0.4	63.49±0.8	69.83±0.5	68.02±0.8	74.67
RSC [16]	97.61±0.0	61.86±0.0	73.93±0.0	68.32±0.0	75.43
Ours	**98.12±0.1**	**66.80±0.3**	**74.77±0.4**	**70.43±0.1**	**77.53**

Method	Caltech	LabelMe	Pascal	Sun	Avg.
Deep All	96.45	60.03	70.41	62.63	72.38
1-NN	96.51	61.44	71.82	63.46	73.31
Ours (G_ϕ = GAN)	97.89	**67.18**	74.59	70.28	77.48
Ours (G_ϕ = VAE)	**98.12**	66.80	**74.77**	**70.43**	**77.53**

Pandey et al. “Generalization on Unseen Domains via Inference-time Label-Preserving Target Projections” (2020)

Sample Adaptation

$$\mathbf{x}'_s = \psi(\mathbf{x}_t), \quad \mathbf{y}_t = f_{\boldsymbol{\theta}_s}(\mathbf{x}'_s).$$

Feature adjustment

- Learned to extract useful information
- Efficiente and easy In feature space

Generative models

Input adjustment

- Advances in generative models
- Efficiente and easy In feature space

Xiao e Snoek “Beyond Model Adaptation at Test Time: A Survey”

Sample Adaptation

$$\mathbf{x}'_s = \psi(\mathbf{x}_t), \quad \mathbf{y}_t = f_{\boldsymbol{\theta}_s}(\mathbf{x}'_s).$$

Feature adjustment

- Learned to extract useful information
- Efficiente and easy In feature space

Generative models

- No extra data or fine tuning
- Unaffected by number of data
- More stable

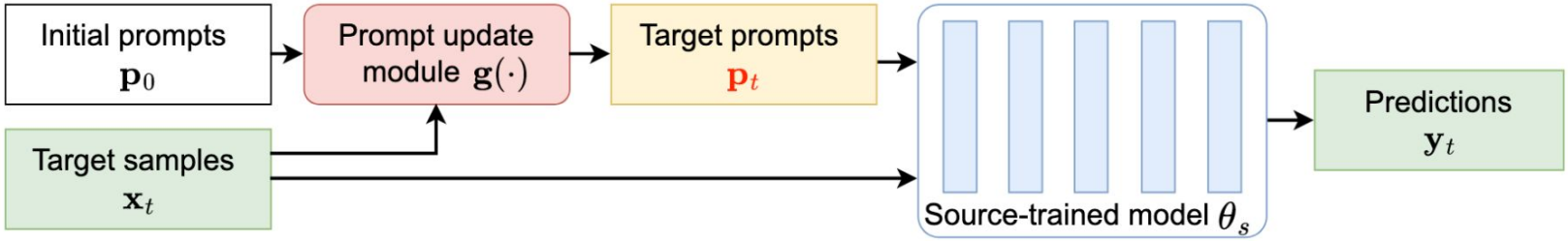
Input adjustment

- Advances in generative models
- Efficiente and easy In feature space

Iterative updates of sample
-> worse efficiency

Xiao e Snoek “Beyond Model Adaptation at Test Time: A Survey”

Prompt Adaptation



$$\mathbf{p}_t = \mathbf{g}(\mathbf{x}_t, \mathbf{p}_0), \quad \mathbf{y}_t = f_{\boldsymbol{\theta}}(\mathbf{x}_t, \mathbf{p}_t)$$

Text-prompting

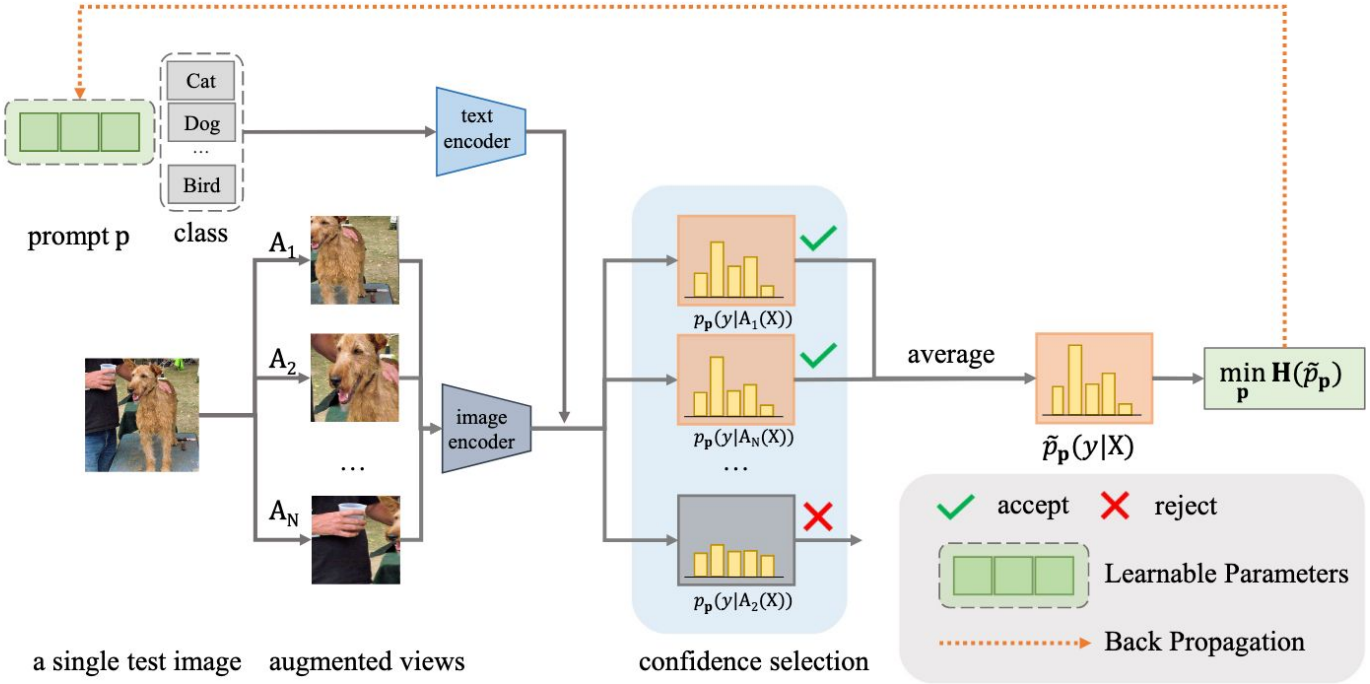
- Prompt-drive approaches
- Use LLMs to provide description or concepts

Embedding-prompting

- Leverage Transformer and VLMs

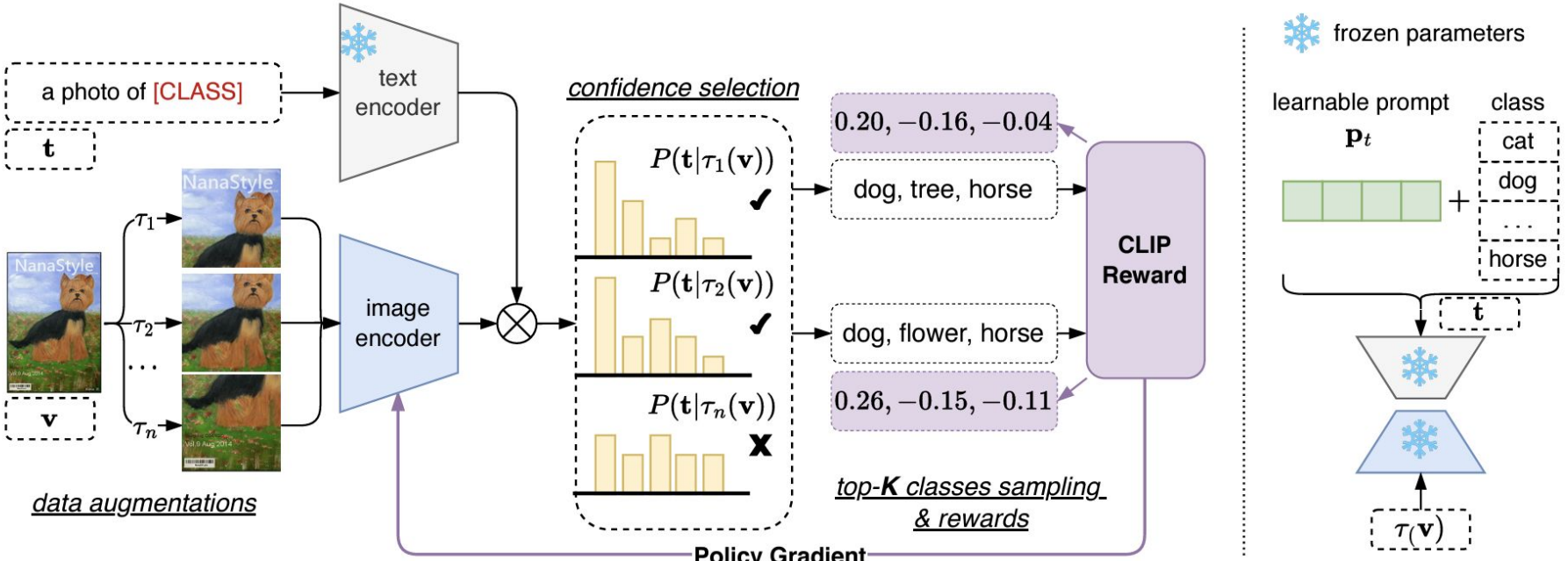
Xiao e Snoek “Beyond Model Adaptation at Test Time: A Survey”

Prompt Adaptation



Shu et al. “Test-Time Prompt Tuning for Zero-Shot Generalization in Vision-Language Models”

Prompt Adaptation



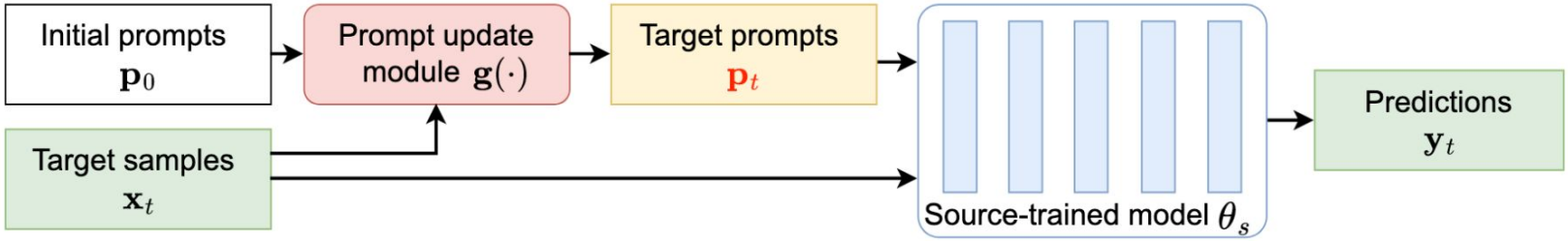
Zhao et al. “Test-Time Adaptation with CLIP Reward for Zero-Shot Generalization in Vision-Language Models”

Prompt Adaptation

Method	ImageNet	ImageNet-A	ImageNet-V2	ImageNet-R	ImageNet-Sketch	OOD Average
			Zero-shot baseline			
CLIP-ViT-B/16	66.73	47.87	60.86	73.98	46.09	57.20
CLIP-ViT-L/14	73.44	68.82	67.80	85.40	57.84	69.97
			Prompt tuning for CLIP-ViT-B/16			
CoOp (Zhou et al., 2021)	71.51	49.71	64.20	75.21	47.99	59.28
CoCoOp (Zhou et al., 2022)	71.02	50.63	64.07	76.18	48.75	59.91
TPT (Manli et al., 2022)	68.98	54.77	63.45	77.06	47.94	60.81
TPT + CoOp (Manli et al.)	73.61	57.95	66.83	77.27	49.29	62.84
TPT + CoOp + KD (Hinton et al.)	71.40	63.25	65.28	82.70	55.78	66.75
RLCF	73.23 (↑6.50)	65.45 (↑17.58)	69.77 (↑8.91)	83.35 (↑9.37)	54.74 (↑8.65)	68.33 (↑11.13)
RLCF + CoOp	76.05 (↑4.54)	69.74 (↑20.03)	70.62 (↑6.42)	84.51 (↑9.30)	56.49 (↑8.50)	70.34 (↑11.06)
RLCF-S + CoOp	76.50 (↑4.99)	71.11 (↑21.40)	70.92 (↑6.72)	84.73 (↑9.52)	56.97 (↑8.98)	70.93 (↑11.65)
			Image encoder tuning for CLIP-ViT-B/16			
Pseudo-label (Lee et al., 2013)	69.11	62.15	63.56	80.03	49.45	63.80
TPT (Manli et al., 2022)	69.42	61.62	63.70	79.74	49.47	63.63
KD (Hinton et al., 2015)	70.92	66.39	65.01	82.12	53.51	66.76
ATKD (Guo et al., 2020)	70.51	70.66	65.54	85.12	53.56	68.72
RLCF	74.85 (↑8.12)	73.71 (↑25.84)	69.77 (↑8.91)	86.19 (↑12.21)	57.10 (↑11.01)	71.69 (↑14.49)
RLCF-S	75.34 (↑8.61)	75.00 (↑27.13)	70.08 (↑9.22)	86.97 (↑12.99)	57.75 (↑11.66)	72.45 (↑15.25)
RLCF-S-M	75.48 (↑8.75)	75.16 (↑27.29)	70.42 (↑9.56)	87.23 (↑13.25)	57.73 (↑11.64)	72.64 (↑15.44)

Zhao et al. “Test-Time Adaptation with CLIP Reward for Zero-Shot Generalization in Vision-Language Models”

Prompt Adaptation

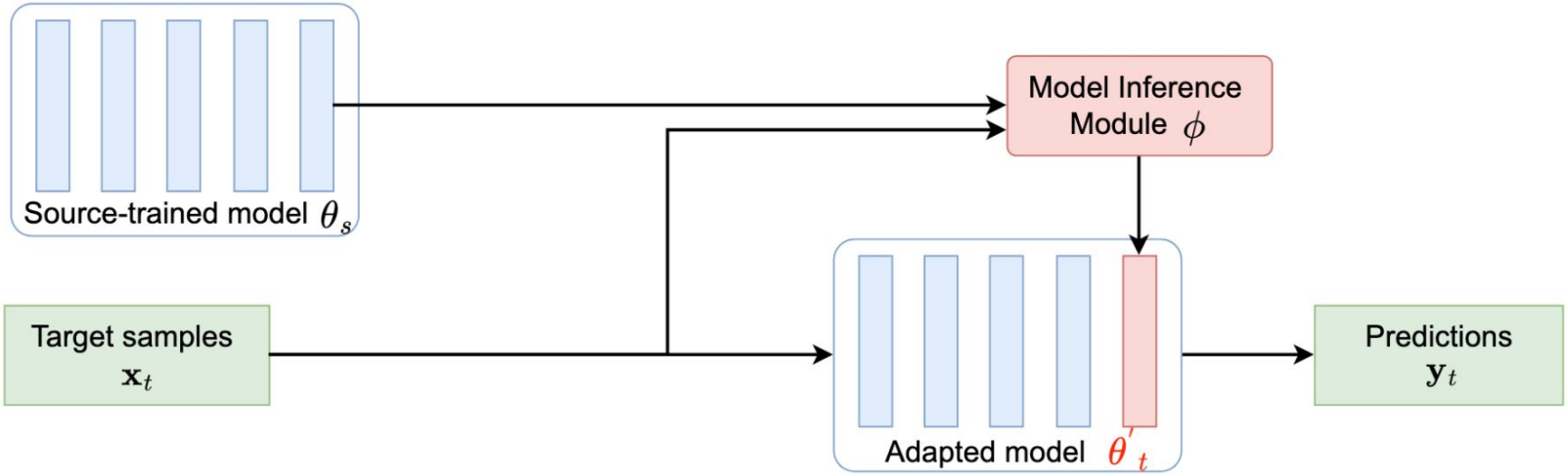


$$\mathbf{p}_t = \mathbf{g}(\mathbf{x}_t, \mathbf{p}_0), \quad \mathbf{y}_t = f_{\boldsymbol{\theta}}(\mathbf{x}_t, \mathbf{p}_t)$$

- may be overconfident
- NLP drawbacks
- task-specific

Xiao e Snoek "Beyond Model Adaptation at Test Time: A Survey"

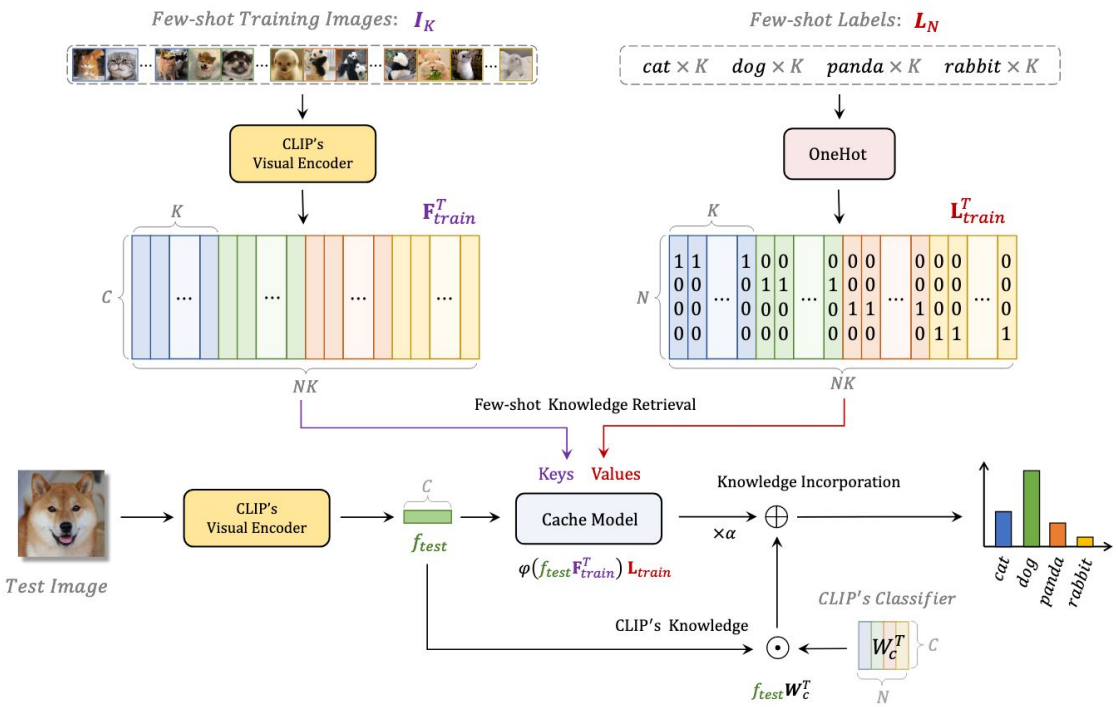
Inference Adaptation

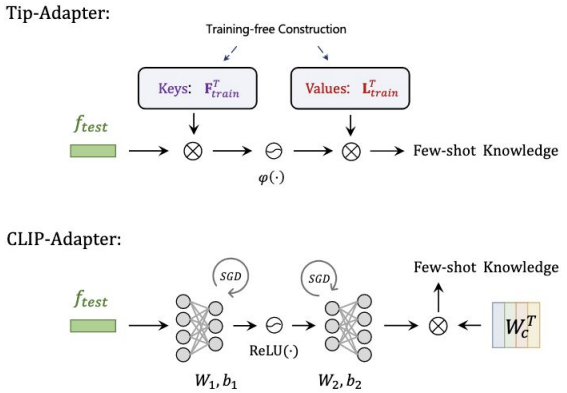


$$\boldsymbol{\theta}'_t = \boldsymbol{\phi}(\mathbf{x}_t, \boldsymbol{\theta}_s), \quad \mathbf{y}_t = f_{\boldsymbol{\theta}'_t}(\mathbf{x}_t)$$

Xiao e Snoek "Beyond Model Adaptation at Test Time: A Survey"

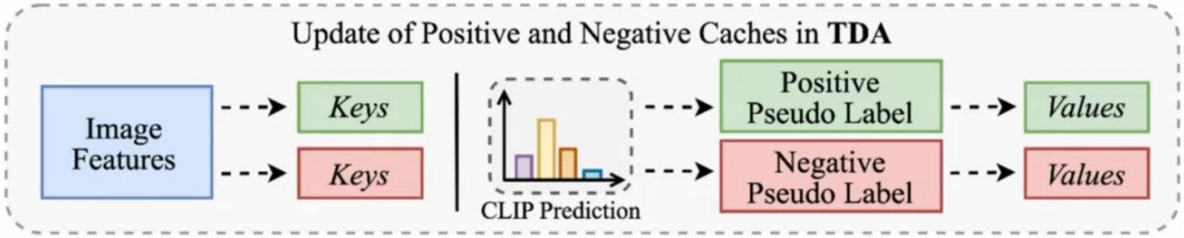
Tip-Adapter





Zhang et al. “Tip-Adapter: Training-free Adaption of CLIP for Few-shot Classification”

Training-free Dynamic Adapter (TDA)



- Cache Update Mechanism:
  Both key-value caches update dynamically, integrating test features and corresponding pseudo labels from CLIP predictions, based on prediction entropy and cache capacity

Positive Cache

- **Capacity:**
  Holds up to **k** high-confidence samples per class
- **Impact**:
  Continuously improves testing accuracy by integrating the most reliable samples

Negative Cache

- **Capacity:**
  Contains up to **k** samples per class, focused on uncertain predictions
- **Impact**:
  Actively reduces prediction errors by managing and adjusting uncertain pseudo labels

Karmanov et al. “Efficient Test-Time Adaptation of Vision-Language Models

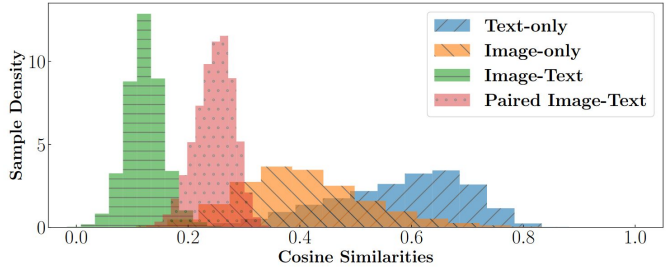
Proposal 1: Create a neutral cache

- Idea: Intermediate cache between positive and negative caches.

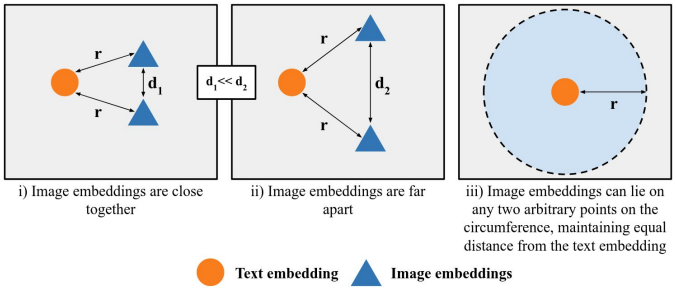
$$P_{\text{TDA}}(f_{\text{test}}) = f_{\text{test}}\mathbf{W}_c^T + P_{\text{pos}}(f_{\text{test}}) + P_{\text{neg}}(f_{\text{test}})$$

$$P_{\text{TDA}}(f_{\text{test}}) = f_{\text{test}}W_c^T + P_{\text{pos}}(f_{\text{test}}) + P_{\text{neg}}(f_{\text{test}}) + P_{\text{neut}}(f_{\text{test}})$$

Proposal 2: Tackle intra-modality gap with TIP-X



(a) **Intra-modal and inter-modal CLIP cosine similarities.** We observe quite distinct intra-modal and inter-modal cosine similarity distributions.



(b) **Intra-modal degrees of freedom**. Different intra-modal similarities can satisfy same inter-modal constraints, leaving room for poor calibration.

Idea: Compute similarities between image features and text to decrease intra-modality gap.

What about using support sets obtained from LAION-5B?

Proposal 2: Tackle intra-modality gap with TIP-X

C: number of classes
K: number of examples of classes in our cache

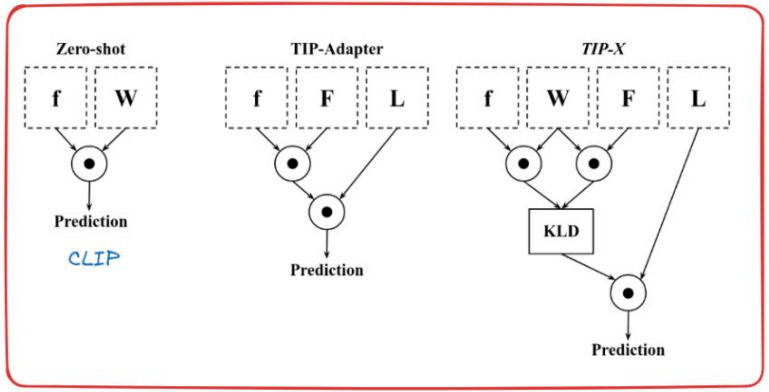
$$S = \texttt{softmax}(FW^T), S \in \mathbb{R}^{CK \times C}$$

$$s = \texttt{softmax}(fW^T), s \in \mathbb{R}^{t \times C}$$

$$M_{i,j} = \texttt{KL}(s_i || S_j), i \in [1, t], j \in [1, CK]$$

$$\texttt{KL}(P||Q) = \sum_i P_i \log \frac{P_i}{Q_i}$$

discrete probabilities -> discrete KL-div



$$\texttt{TXL} = fW^T + \alpha AL + \gamma \psi(-M)L$$

TIP-X Logits

Rescales M to have same min max values as the TIP-Adapter affinities

inverts similarity meaning

Thank you for your attention

Questions?