BOIL: TOWARDS REPRESENTATION CHANGE FOR FEW-SHOT LEARNING

Jaehoon Oh*[1], Hyungjun Yoo*[1], ChangHwan Kim[1] & Se-Young Yun[2]
[1]Graduate School of Knowledge Service Engineering, KAIST
[2]Graduate School of Artificial Intelligence, KAIST
{jaehoon.oh,yoohjun,kimbob,yunseyoung}@kaist.ac.kr

2021.07.01
임진혁

Index

1. Introduction

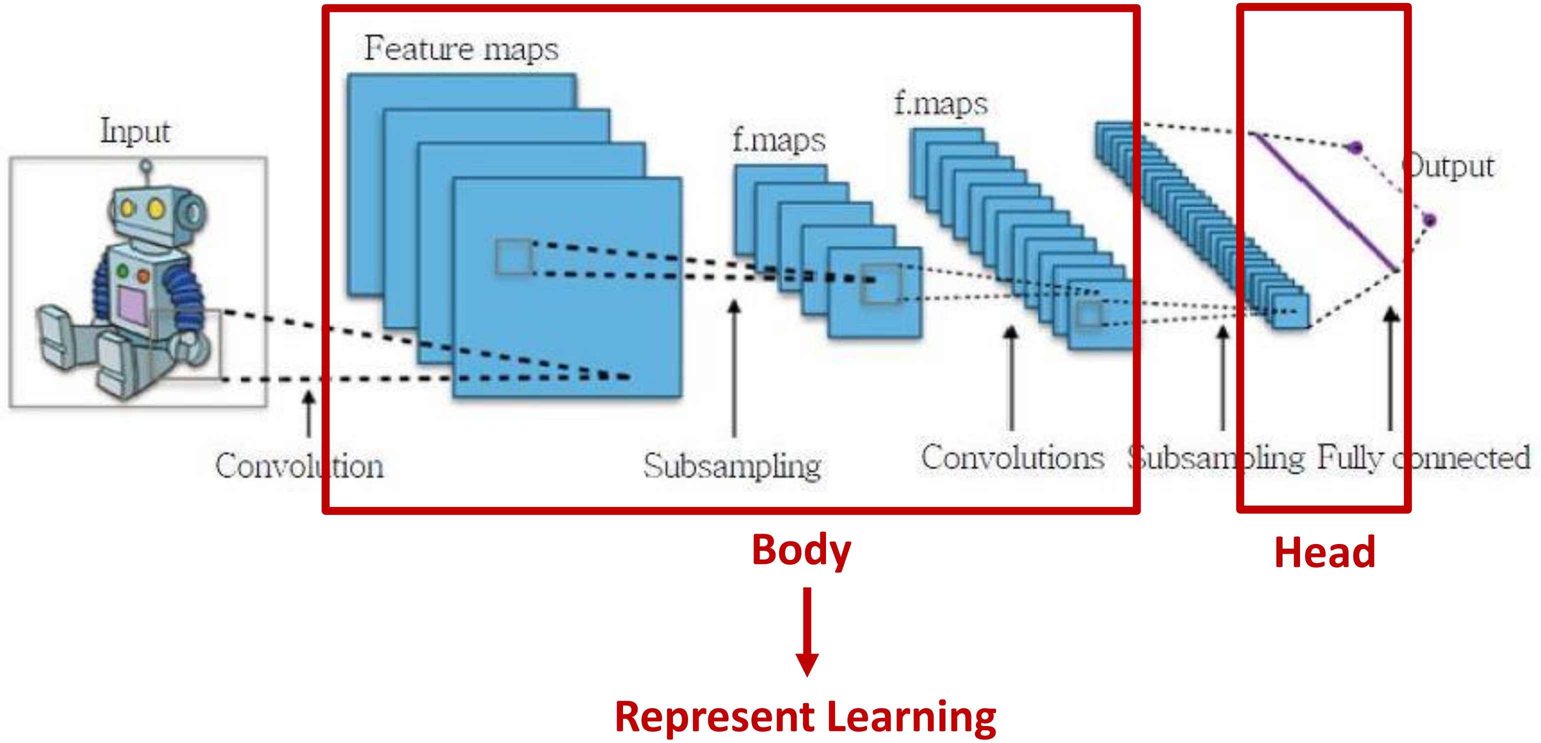
2. Proposed Approach

Introduction

Recent studys hypothesized that,
“MAML’s success is from **High Quality features** before inner updates”
(= ***“represent reuse”***)

This Paper say,
“For ultimate purpose of Meta Learning,
it should be not ***‘represent reuse’*** but ***‘represent change’ “***

임진혁<eeplearning@units.ac.kr>

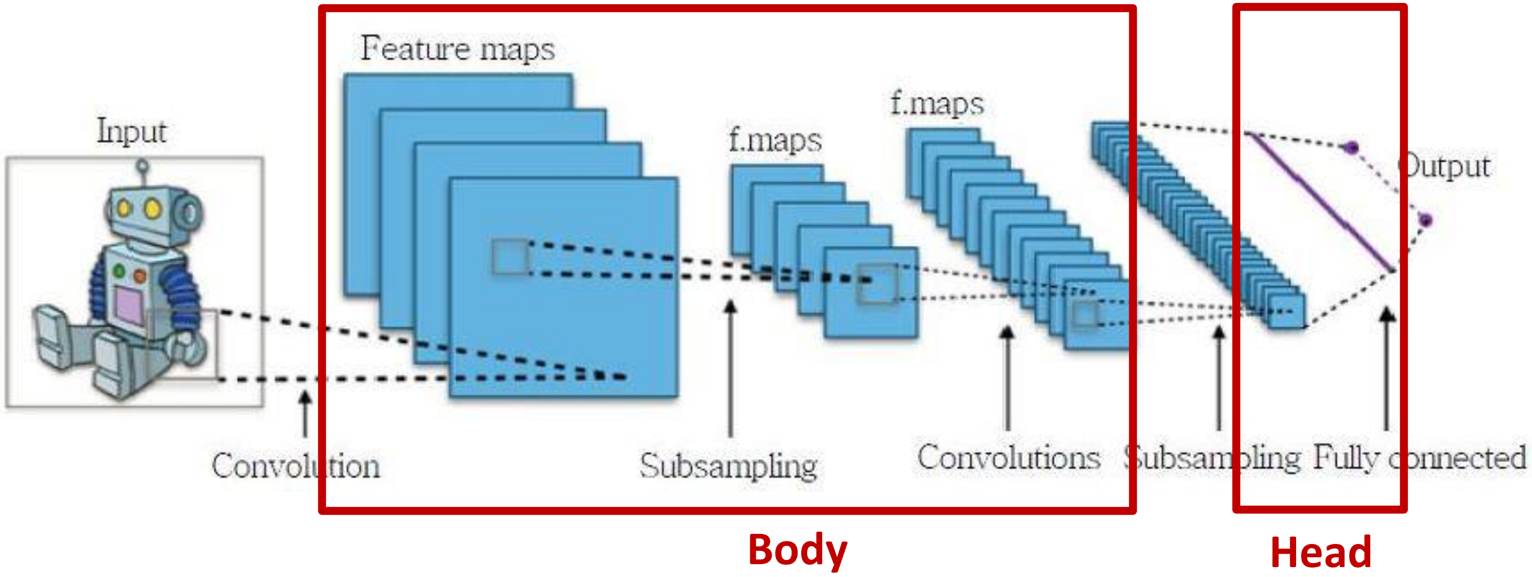


Rapid learning or feature reuse? towards understanding the effectiveness of maml (ICLR2020) **이하 reuse paper**

MAML learns new tasks by updating the head with almost **same features.**

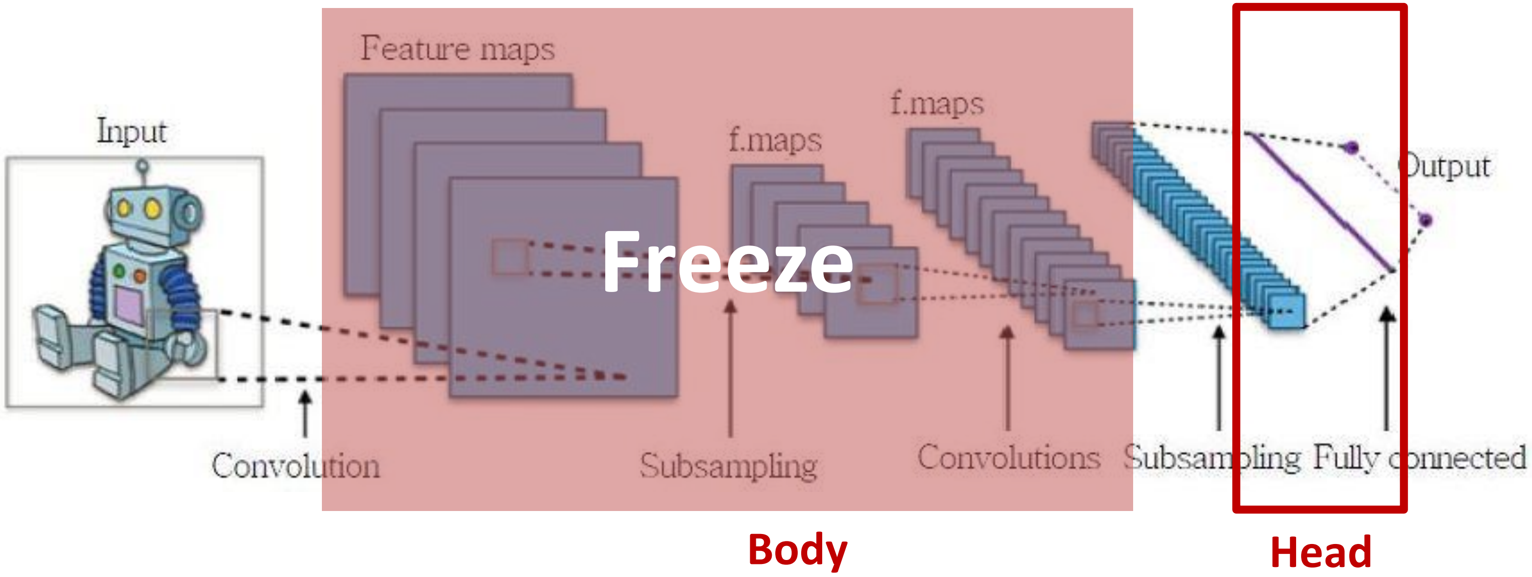
represent reuse

Introduction



represent reuse: small changes in the representations during task learning
represent change: large changes in the representations during task learning

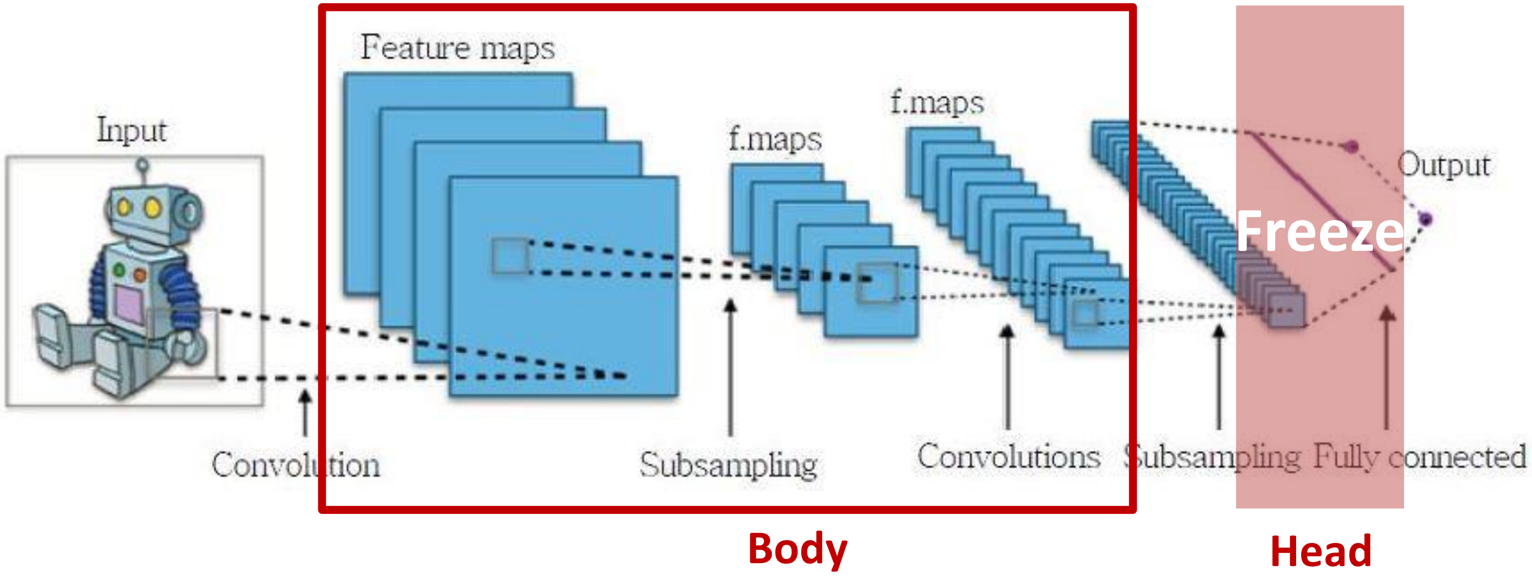
1 Introduction



ANIL (Almost No Inner Loop) freeze Body but only update Head during inner update
But there is no significant performance loss.

represent reuse: small changes in the representations during task learning

Introduction

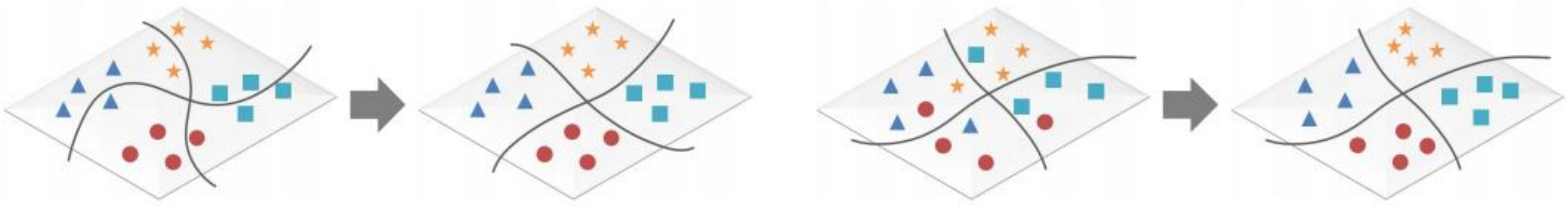


In this paper, BOIL(Body Only update in Inner Loops) is suggested as suitable for ultimate purpose of Meta Learning : good at **Different** unseen task

represent change: large changes in the representations during task learning

임진혁<eeplearning@units.ac.kr>

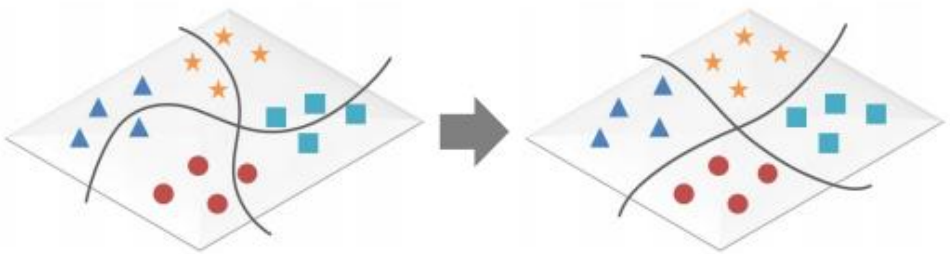
Introduction



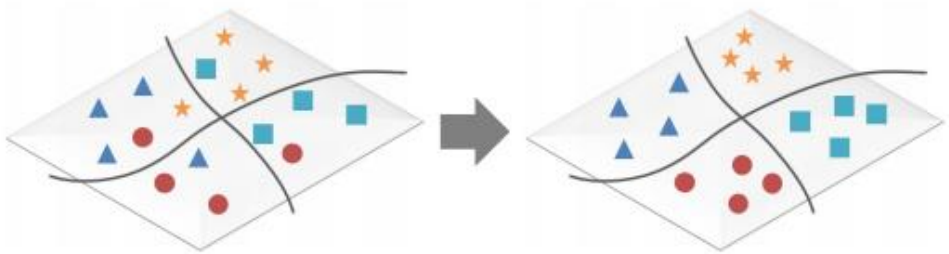
(a) MAML/ANIL. (b) BOIL.

Domain	General (Coarse-grained)		Specific (Fine-grained)	
Dataset	miniImageNet	tieredImageNet	Cars	CUB
MAML(1)	47.44 ± 0.23	47.44 ± 0.18	45.27 ± 0.26	56.18 ± 0.37
ANIL(1)	47.82 ± 0.20	**49.35** ± 0.26	46.81 ± 0.24	57.03 ± 0.41
BOIL(1)	**49.61** ± 0.16	48.58 ± 0.27	**56.82** ± 0.21	**61.60** ± 0.57
MAML(5)	61.75 ± 0.42	64.70 ± 0.14	53.23 ± 0.26	69.66 ± 0.03
ANIL(5)	63.04 ± 0.42	65.82 ± 0.12	61.95 ± 0.38	70.93 ± 0.28
BOIL(5)	**66.45** ± 0.37	**69.37** ± 0.12	**75.18** ± 0.21	**75.96** ± 0.17

1 Introduction



(a) MAML/ANIL.

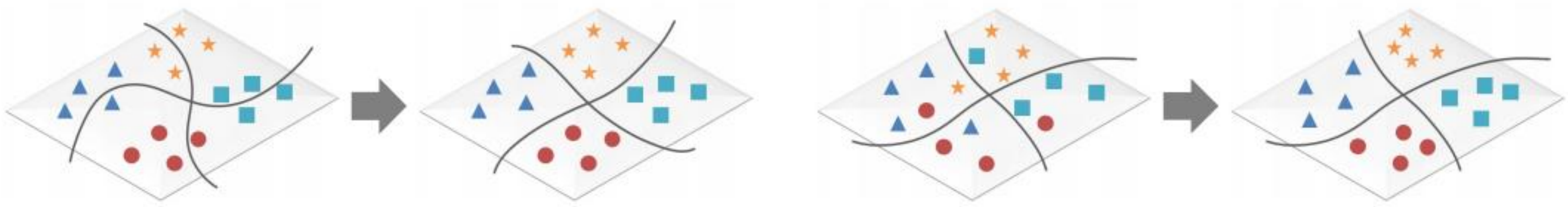


(b) BOIL.

Table 2: Test accuracy (%) of 4conv network on cross-domain adaptation.

adaptation	General to General		General to Specific		Specific to General		Specific to Specific	
meta-train	tieredImageNet	miniImageNet	miniImageNet	miniImageNet	Cars	Cars	CUB	Cars
meta-test	miniImageNet	tieredImageNet	Cars	CUB	miniImageNet	tieredImageNet	Cars	CUB
MAML(1)	47.60 ± 0.24	51.61 ± 0.20	33.57 ± 0.14	40.51 ± 0.08	26.95 ± 0.15	28.46 ± 0.18	32.22 ± 0.30	29.64 ± 0.19
ANIL(1)	49.67 ± 0.31	52.82 ± 0.29	34.77 ± 0.31	41.12 ± 0.15	28.67 ± 0.17	29.41 ± 0.19	33.07 ± 0.43	28.32 ± 0.32
BOIL(1)	**49.74** ± 0.26	**53.23** ± 0.41	**36.12** ± 0.29	**44.20** ± 0.15	**33.71** ± 0.13	**34.06** ± 0.20	**35.44** ± 0.46	**34.79** ± 0.27
MAML(5)	65.22 ± 0.20	65.76 ± 0.27	44.56 ± 0.21	53.09 ± 0.16	30.64 ± 0.19	32.62 ± 0.21	41.24 ± 0.21	32.18 ± 0.13
ANIL(5)	66.47 ± 0.16	66.52 ± 0.28	46.55 ± 0.29	55.82 ± 0.21	35.38 ± 0.10	36.94 ± 0.10	43.05 ± 0.23	37.99 ± 0.15
BOIL(5)	**69.33** ± 0.19	**69.37** ± 0.23	**50.64** ± 0.22	**60.92** ± 0.11	**44.51** ± 0.25	**46.09** ± 0.23	**47.30** ± 0.22	**45.91** ± 0.28

Introduction

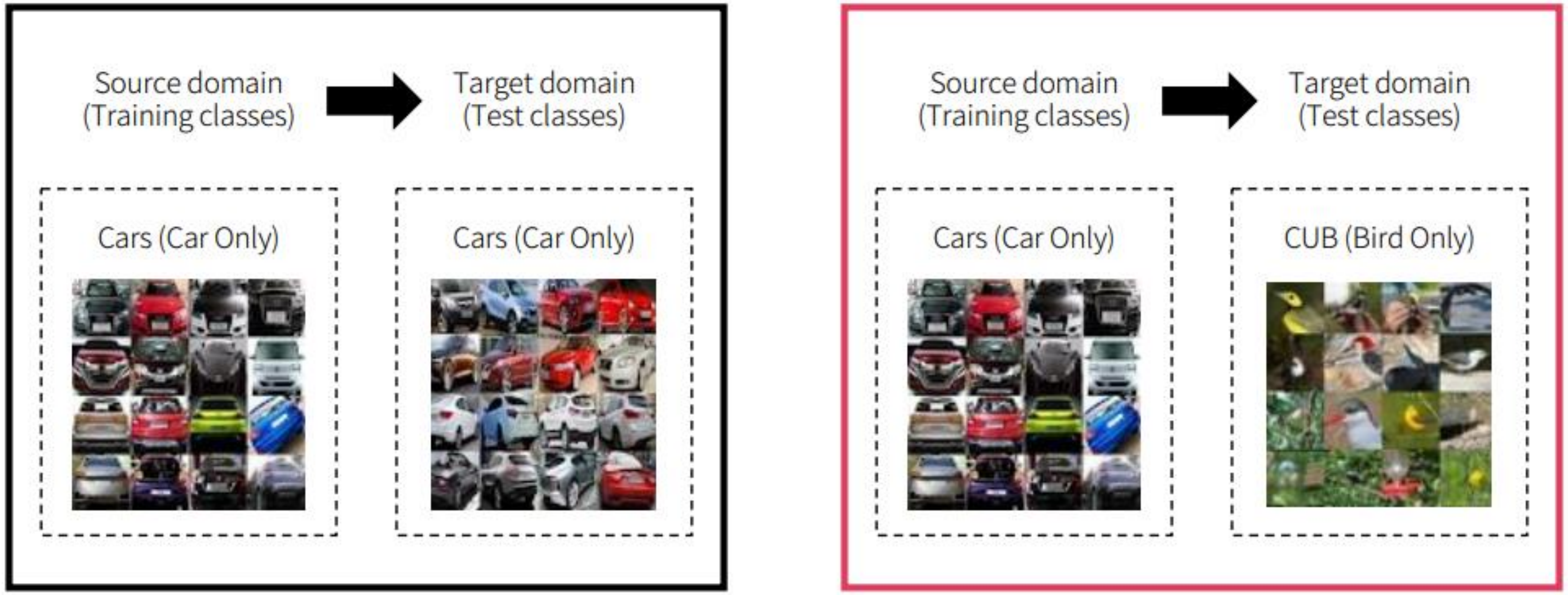


(a) MAML/ANIL. (b) BOIL.

즉, Meta train과 Meta test가 다른 경우 (Cross-Domain)
Meta Learning(MAML)이 잘하기 위해서는
기존의 represent reuse가 아닌 represent change가 필요함을 보여주는 연구

임진혁<eeplearning@units.ac.kr>

Introduction

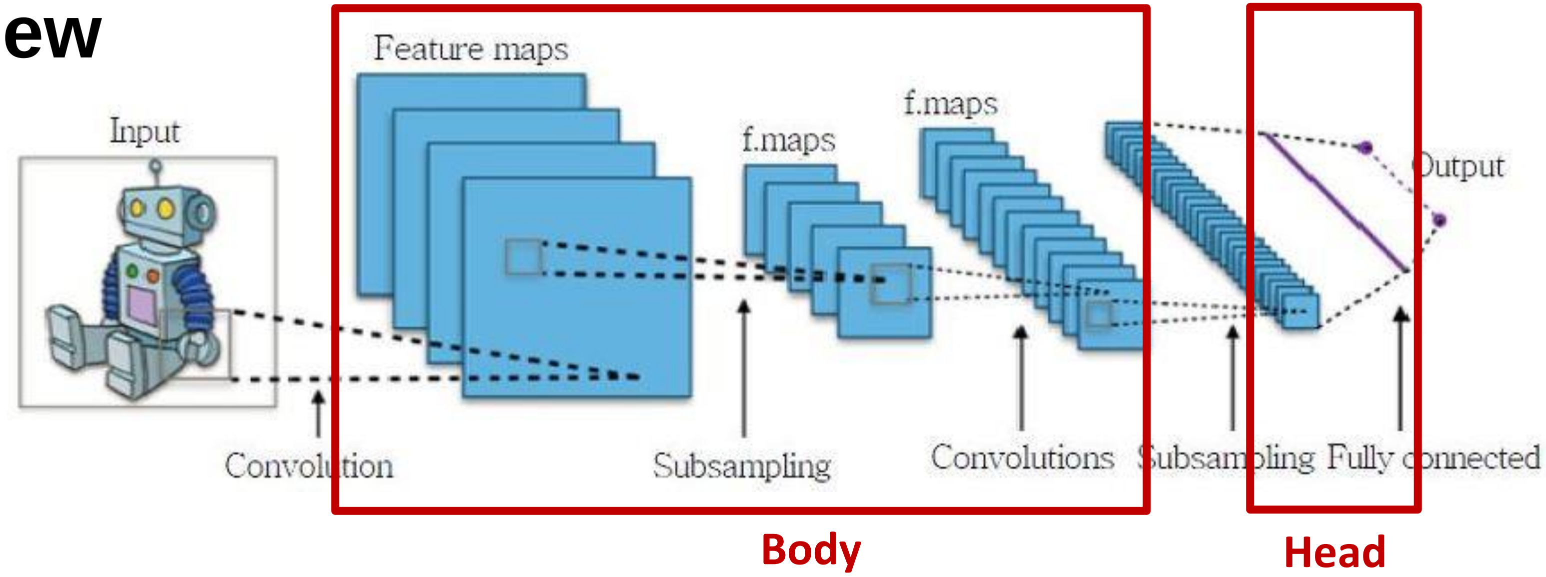


(a) Same domain adaptation example (Unseen, Similar)

(b) Cross domain adaptation example (Unseen, Dissimilar)

즉, Meta train과 Meta test가 다른 경우 (Cross-Domain)
Meta Learning(MAML)이 잘하기 위해서는
기존의 represent reuse가 아닌 represent change가 필요함을 보여주는 연구

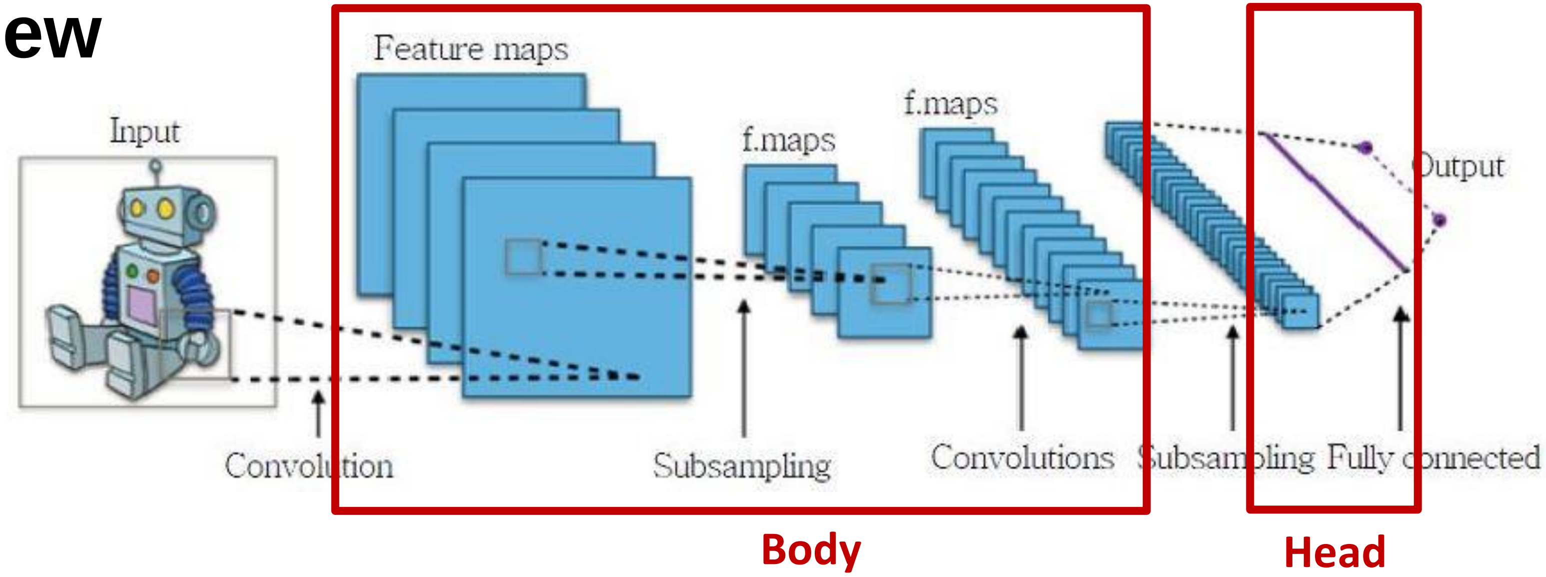
CNN Review



Body: representation Learning
Head: linear decision boundary Learning

임진혁<eeplearning@units.ac.kr>

CNN Review



Body: feature extractor
Head: classifier

MAML Review

Algorithm MAML

Sample a mini-batch of tasks τ_t from $p(\tau)$

for each task $\tau \in \tau_t$ **do**

$$\theta_\tau \leftarrow \theta_0 - \alpha \nabla_{\theta_0} \mathcal{L}(f_{\theta_0}; D^{\tau, sup}_{meta-train})$$

(inner loop, fast adaptation)

end for

$$\theta_0 \leftarrow \theta_0 - \beta \nabla_{\theta_0} \sum_{\tau \in \tau_t} \mathcal{L}(f_{\theta_\tau}; D^{\tau, qry}_{meta-train})$$

(outer loop, meta-initialization)

(a) Meta training using MAML

- Inner Loop: 특정 task에 대한 initial parameter에서부터 시작해서 성능을 높이는 것
- Outer Update: 좋은 initial point을 찾는 것

임진혁<eeplearning@units.ac.kr>

MAML Review

Algorithm MAML

Sample a mini-batch of tasks τ_t from $p(\tau)$

for each task $\tau \in \tau_t$ do

$$\theta_\tau \leftarrow \theta_0 - \alpha \nabla_{\theta_0} \mathcal{L}(f_{\theta_0}; D^{\tau, sup}_{meta-train})$$

(inner loop, fast adaptation)

end for

$$\theta_0 \leftarrow \theta_0 - \beta \nabla_{\theta_0} \sum_{\tau \in \tau_t} \mathcal{L}(f_{\theta_\tau}; D^{\tau, qry}_{meta-train})$$

(outer loop, meta-initialization)

(a) Meta training using MAML

- Inner Loop: Task Specific Learning
- Outer Update: Universe Learing

임진혁<eeplearning@units.ac.kr>

MAML Review

Algorithm MAML

Sample a mini-batch of tasks τ_t from $p(\tau)$

for each task $\tau \in \tau_t$ do

$$\theta_\tau \leftarrow \theta_0 - \alpha \nabla_{\theta_0} \mathcal{L}(f_{\theta_0}; D^{\tau, sup}_{meta-train})$$

(inner loop, fast adaptation)

end for

$$\theta_0 \leftarrow \theta_0 - \beta \nabla_{\theta_0} \sum_{\tau \in \tau_t} \mathcal{L}(f_{\theta_\tau}; D^{\tau, qry}_{meta-train})$$

(outer loop, meta-initialization)

(a) Meta training using MAML

- Inner Loop: Task Specific Learning
- Outer Update: Universe Learing

임진혁<eeplearning@units.ac.kr>

ANIL vs BOIL

Algorithm MAML

Sample a mini-batch of tasks τ_t from $p(\tau)$

for each task $\tau \in \tau_t$ do

$$\theta_\tau \leftarrow \theta_0 - \alpha \nabla_{\theta_0} \mathcal{L}(f_{\theta_0}; D^{\tau, sup}_{meta-train})$$

(inner loop, fast adaptation)

end for

$$\theta_0 \leftarrow \theta_0 - \beta \nabla_{\theta_0} \textstyle\sum_{\tau \in \tau_t} \mathcal{L}(f_{\theta_\tau}; D^{\tau, qry}_{meta-train})$$

(outer loop, meta-initialization)

(a) Meta training using MAML

$$\theta = \{\theta_b, \theta_h\}$$

Output can be expressed as (In MAML)

$$\hat{y} = f_\theta(x) = f_{\theta_h}(f_{\theta_b}(x))$$

$$\theta_{b,\tau_i} = \theta_b - \alpha_b \nabla_{\theta_b} L_{S_{\tau_i}}(f_\theta)$$

$$\theta_{h,\tau_i} = \theta_h - \alpha_h \nabla_{\theta_h} L_{S_{\tau_i}}(f_\theta)$$

ANIL vs BOIL

Algorithm MAML

Sample a mini-batch of tasks τ_t from $p(\tau)$

for each task $\tau \in \tau_t$ **do**

$$\theta_\tau \leftarrow \theta_0 - \alpha \nabla_{\theta_0} \mathcal{L}(f_{\theta_0}; D^{\tau, sup}_{meta-train})$$

(inner loop, fast adaptation)

end for

$$\theta_0 \leftarrow \theta_0 - \beta \nabla_{\theta_0} \sum_{\tau \in \tau_t} \mathcal{L}(f_{\theta_\tau}; D^{\tau, qry}_{meta-train})$$

(outer loop, meta-initialization)

(a) Meta training using MAML

$$\theta = \{\theta_b, \theta_h\}$$

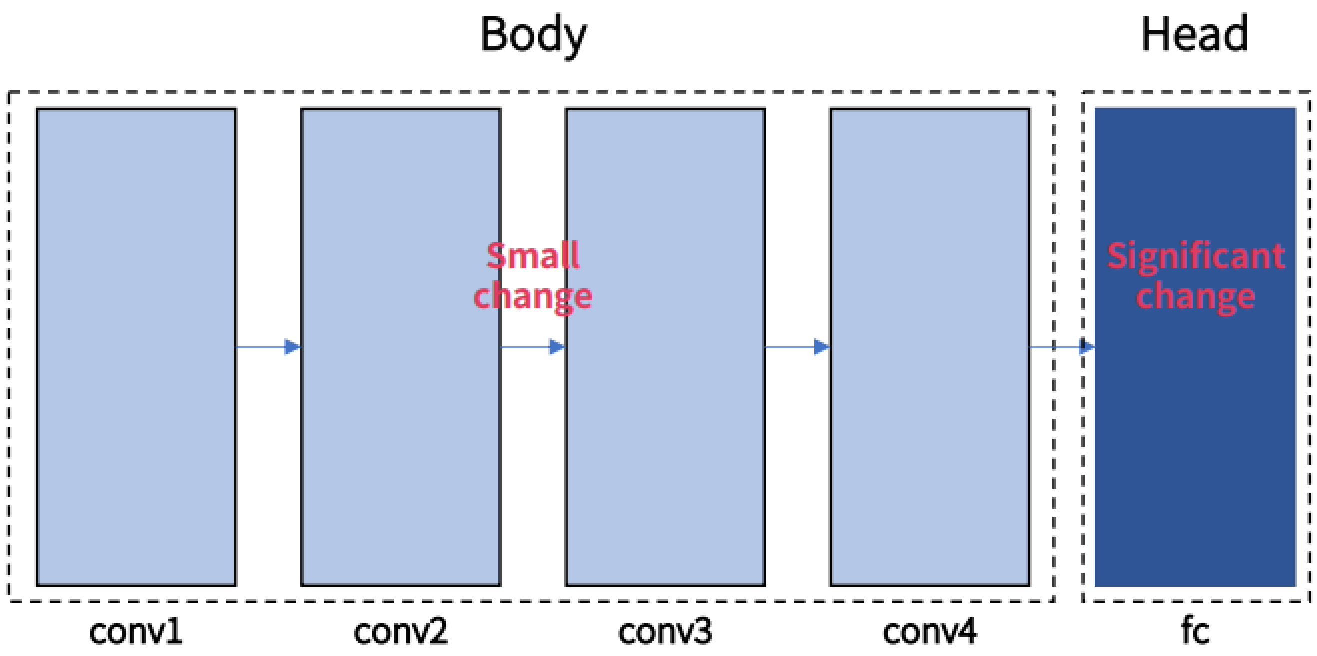
$$\theta_{b,\tau_i} = \theta_b - \alpha_b \nabla_{\theta_b} L_{S_{\tau_i}}(f_\theta)$$

$$\theta_{h,\tau_i} = \theta_h - \alpha_h \nabla_{\theta_h} L_{S_{\tau_i}}(f_\theta)$$

- MAML : $\alpha = \alpha_b = \alpha_h (\neq 0)$
- ANIL: $\alpha_b = 0$ and $\alpha_h \neq 0$
- BOIL: $\alpha_b \neq 0$ and $\alpha_h = 0$

임진혁<eeplearning@units.ac.kr>

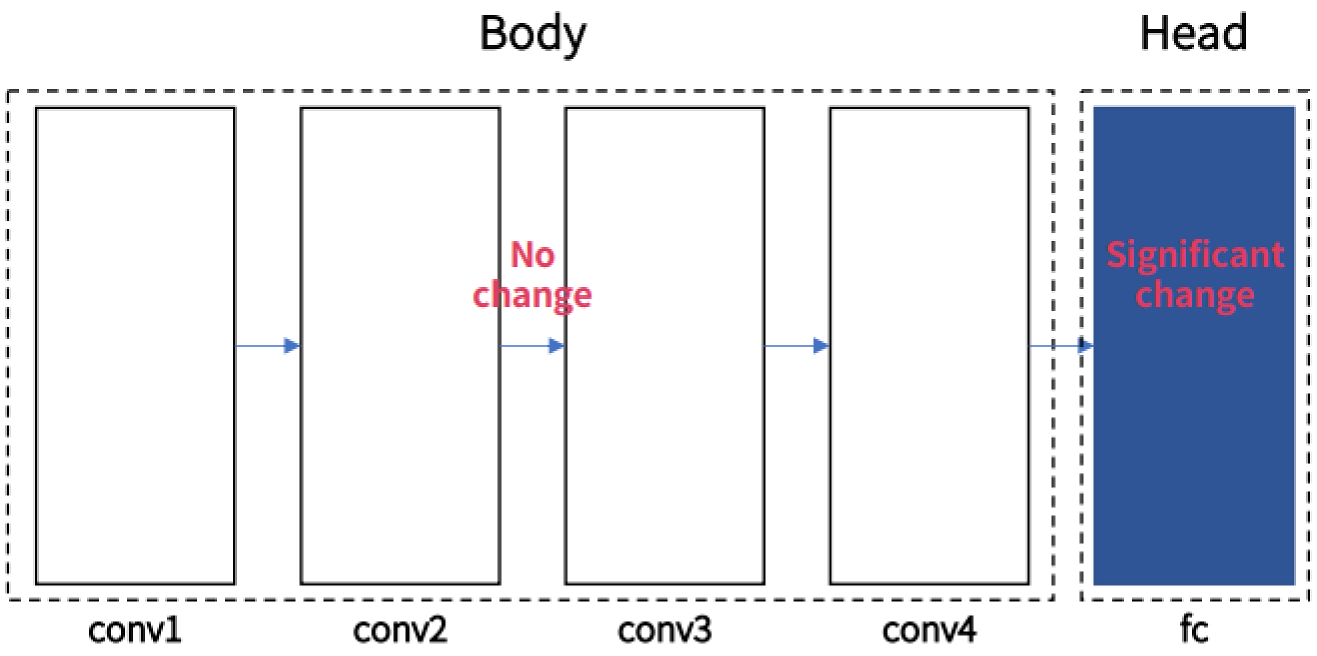
ANIL (force MAML to strength aspect of “represent reuse”)



(a) Changes in MAML through inner loops

임진혁<eeplearning@units.ac.kr>

ANIL (force MAML to strength aspect of “represent reuse”)



(a) Changes in ANIL through inner loops

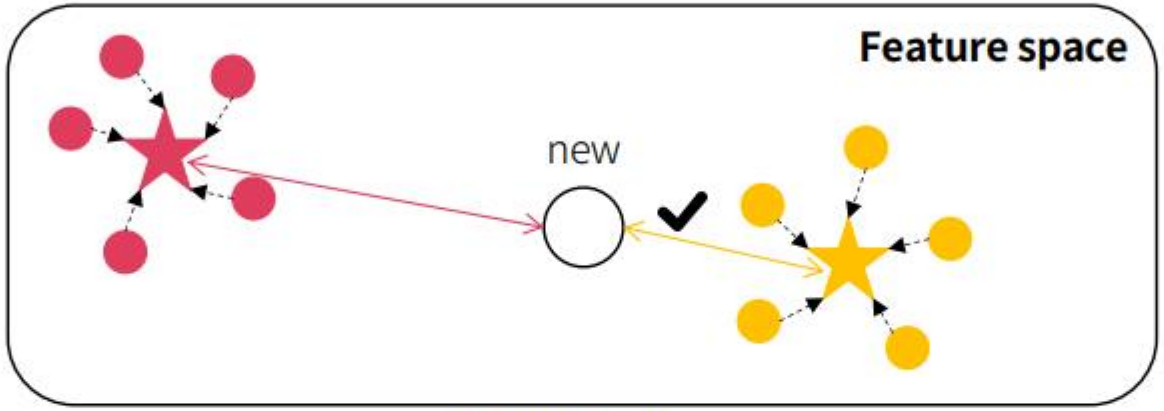
MAML(ANIL) is success to adopt new task quickly (in few steps)

And it can be through sufficient good high features (initial point)

(This paper says it as “represent reuse”)

임진혁<eeplearning@units.ac.kr>

MAML(ANIL) is success to adopt new task quickly (in few steps)
And it can be through sufficient good high features (initial point)
Let's see NIL Test (No Inner Loop Test)



(a) NIL testing

Method	MiniImageNet-5way-1shot	MiniImageNet-5way-5shot
MAML training-NIL head	48.4 ± 0.3	61.5 ± 0.8
ANIL training-NIL head	48.0 ± 0.7	62.2 ± 0.5

임진혁<eeplearning@units.ac.kr>

엥 그러면 BOIL은 왜 필요해요?
ANIL 방식이 원래 MAML의 성공 이유 이기도 하고
성능도 더 증가하는데??

메타 러닝의 본질적인 목적이 무엇일까?

"represent reuse" :

Train Task에서 잘하던 represent 을 Test Task에서 그대로 사용한다?

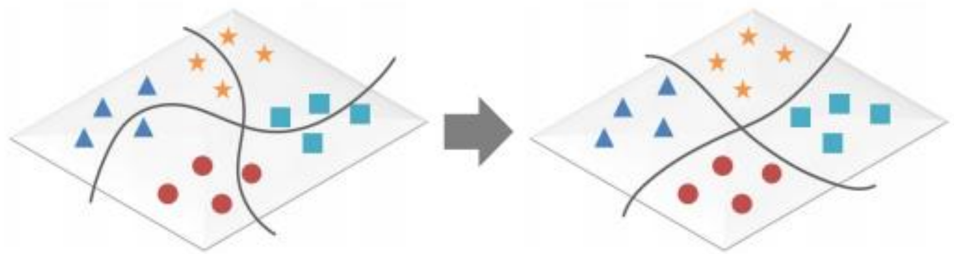
Table 2: Test accuracy (%) of 4conv network on cross-domain adaptation.

adaptation	General to General		General to Specific		Specific to General		Specific to Specific	
meta-train	tieredImageNet	miniImageNet	miniImageNet	miniImageNet	Cars	Cars	CUB	Cars
meta-test	miniImageNet	tieredImageNet	Cars	CUB	miniImageNet	tieredImageNet	Cars	CUB
MAML(1)	47.60 ± 0.24	51.61 ± 0.20	33.57 ± 0.14	40.51 ± 0.08	26.95 ± 0.15	28.46 ± 0.18	32.22 ± 0.30	29.64 ± 0.19
ANIL(1)	49.67 ± 0.31	52.82 ± 0.29	34.77 ± 0.31	41.12 ± 0.15	28.67 ± 0.17	29.41 ± 0.19	33.07 ± 0.43	28.32 ± 0.32
BOIL(1)	**49.74** ± 0.26	**53.23** ± 0.41	**36.12** ± 0.29	**44.20** ± 0.15	**33.71** ± 0.13	**34.06** ± 0.20	**35.44** ± 0.46	**34.79** ± 0.27
MAML(5)	65.22 ± 0.20	65.76 ± 0.27	44.56 ± 0.21	53.09 ± 0.16	30.64 ± 0.19	32.62 ± 0.21	41.24 ± 0.21	32.18 ± 0.13
ANIL(5)	66.47 ± 0.16	66.52 ± 0.28	46.55 ± 0.29	55.82 ± 0.21	35.38 ± 0.10	36.94 ± 0.10	43.05 ± 0.23	37.99 ± 0.15
BOIL(5)	**69.33** ± 0.19	**69.37** ± 0.23	**50.64** ± 0.22	**60.92** ± 0.11	**44.51** ± 0.25	**46.09** ± 0.23	**47.30** ± 0.22	**45.91** ± 0.28

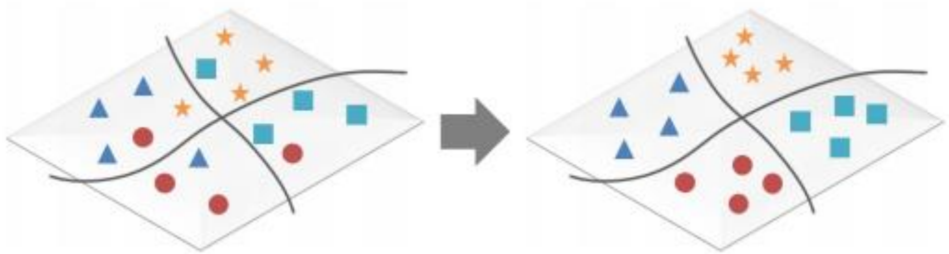
2 Proposed Method

“represent change” :

Task Specific Learning을 Feature Extract (represent) 부분에서 한다.



(a) MAML/ANIL.

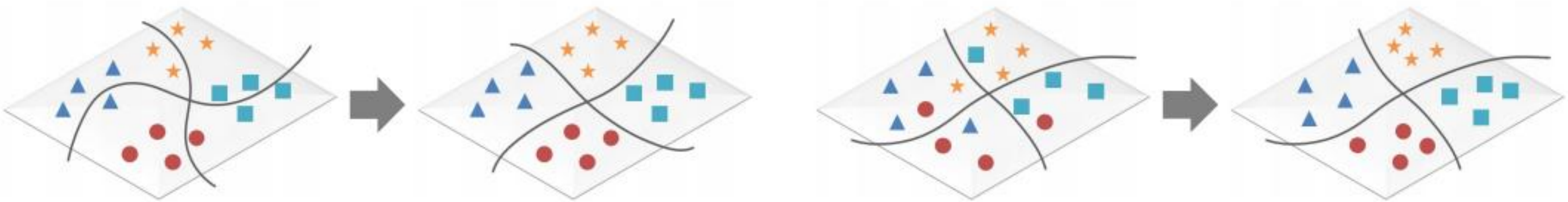


(b) BOIL.

임진혁<eeplearning@units.ac.kr>

"represent change" :

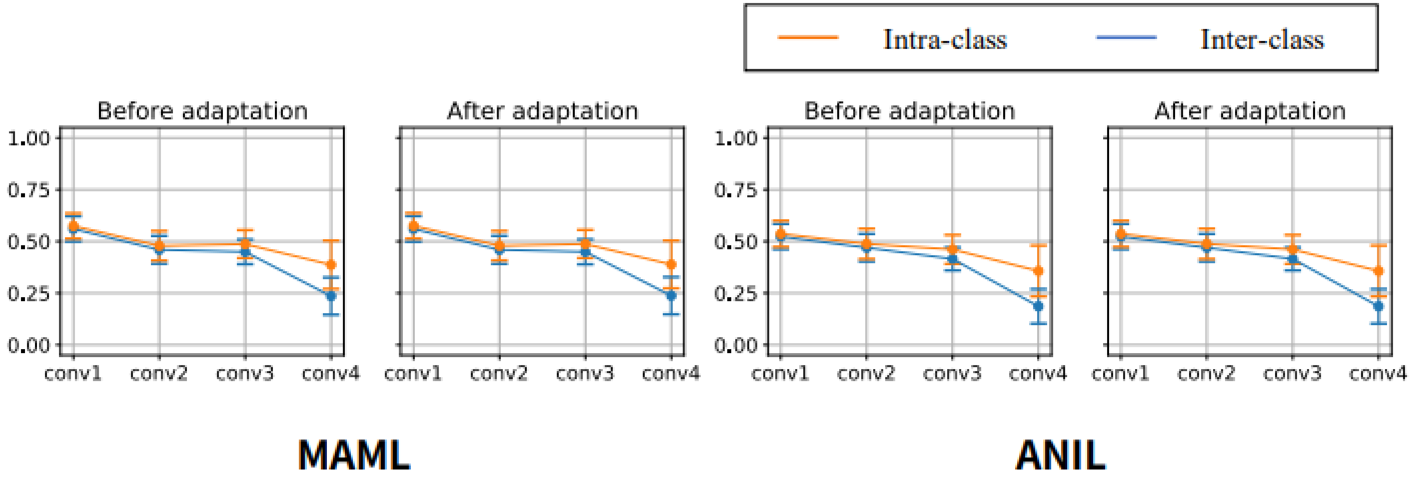
Inner Loop에서 Head를 얼리고 Body만 update 한다. (Body Only IL)



(a) MAML/ANIL. (b) BOIL.

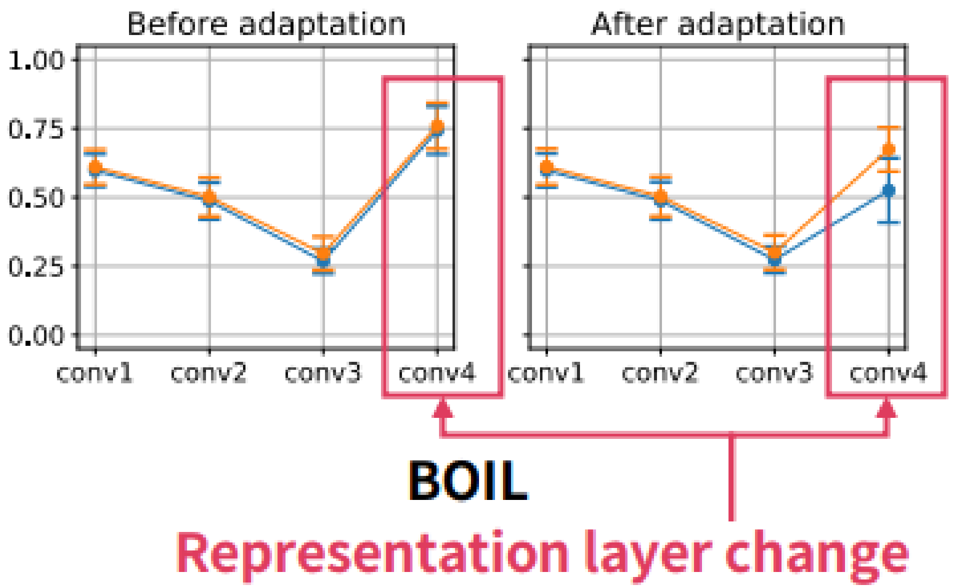
임진혁<eeplearning@units.ac.kr>

2 Proposed Method



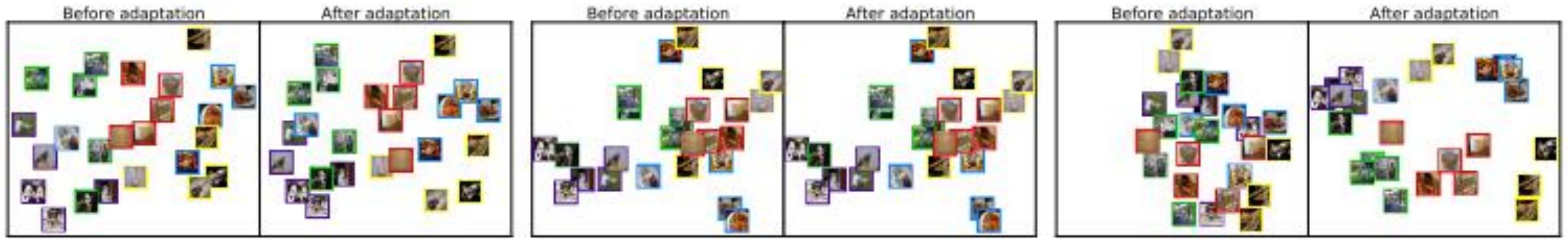
그러니까 원래는 universe(outer update) represent 가 학습 되었는데 (기존 MAML) 해당 Task의 distributio에서 나오는 task는 전부 잘맞출 수 있을 정도로 (head의 decision boundary 학습이 굉장히 적어도(few step) 성능 하락이 없을 정도로)

이제는 task distribution(domain) specific하지 않은, 데이터 자체의 특성(공통적인 feature)을 boundary가 주어졌을 때 최대한 빠르게 boundar를 만족할 수 있는 feature extractor가 된다



2 Proposed Method

B.1 BENCHMARK DATA SETS



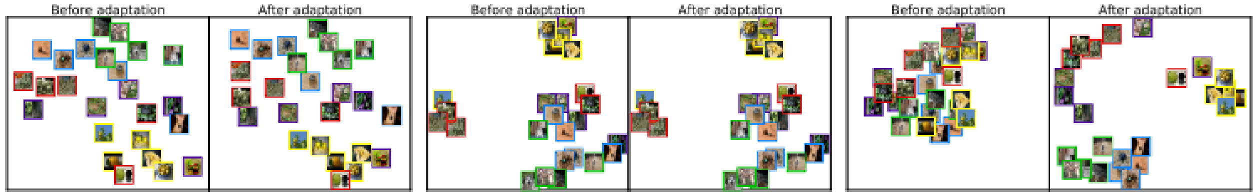
(a) MAML. (b) ANIL. (c) BOIL.

Figure 5: UMAP of samples from miniImageNet using the model meta-trained on miniImageNet.

(a) MAML. (b) ANIL. (c) BOIL.

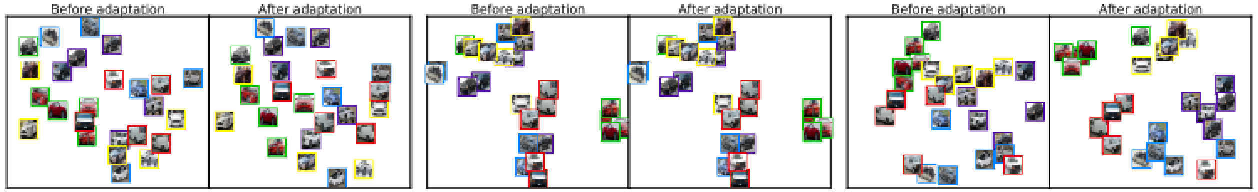
Figure 6: UMAP of samples from Cars using the model meta-trained on Cars.

B.2 Cross-domain adaptation



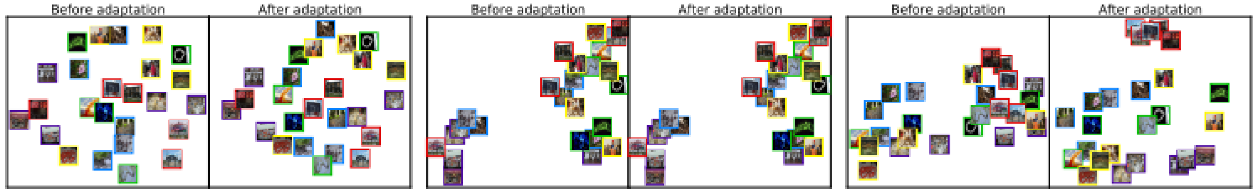
(a) MAML. (b) ANIL. (c) BOIL.

Figure 7: UMAP of samples from tieredImageNet using the model meta-trained on miniImageNet.



(a) MAML. (b) ANIL. (c) BOIL.

Figure 8: UMAP of samples from Cars using the model meta-trained on miniImageNet.



(a) MAML. (b) ANIL. (c) BOIL.

임진혁<eeplearning@units.ac.kr>

Conclusion

MAML은 "represent reuse"로 좋은 성능이었다.

BOIL은 MAML에서 *"represent change"* 을 강화시켰다.

representation change for solving **domain-agnostic tasks** and proposed the BOIL algorithm

임진혁<eeplearning@units.ac.kr>

감사합니다

임진혁<eeplearning@unist.ac.kr>