Meta-Learning of Neural Architectures for Few-Shot Learning

김지욱
Open Seminar(2024.06.04)

School of Intelligent Mechatronics Engineering,
Sejong University

발표자 소개

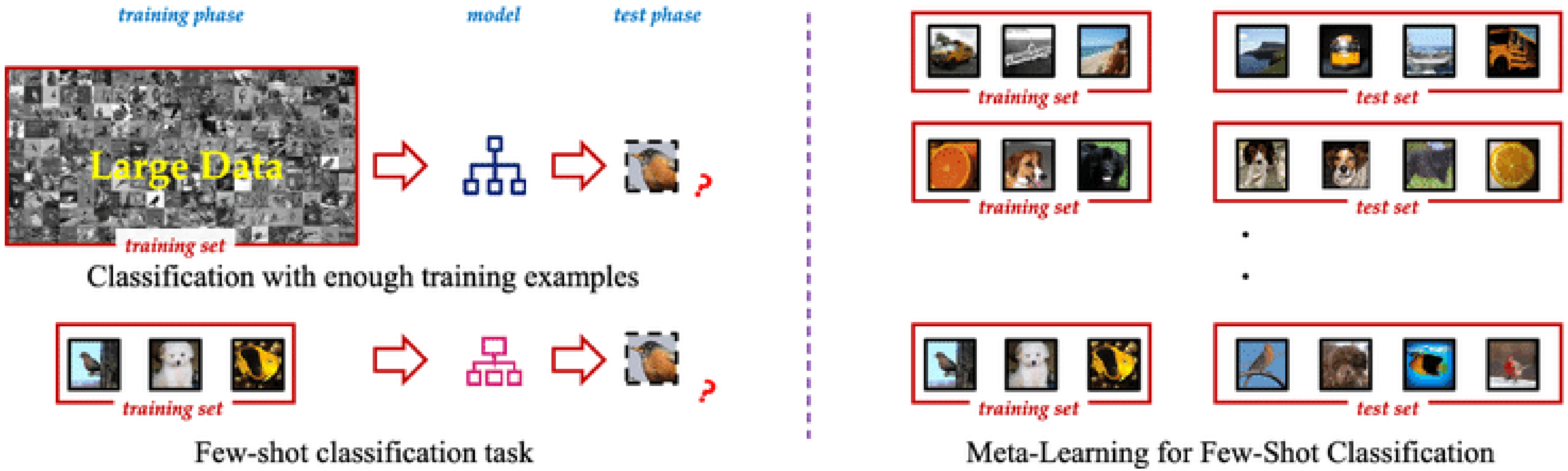
- **김지욱(Jiuk Kim)**
  - 세종대학교 AI로봇학과 재학
  - Machine Intelligence and Networking Lab. (이현석 교수님)

- **Research Interest**
  - AutoML

- **Contact**
  - 22011893@sju.ac.kr

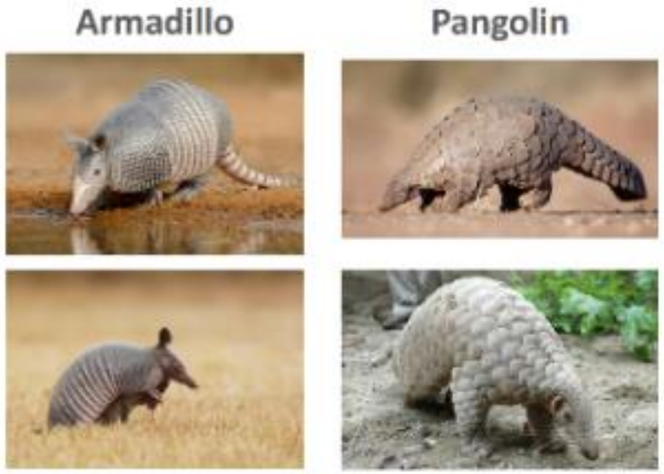
Background

Neural Architecture Search => 많은 데이터와 컴퓨팅 리소스를 필요로 함
=> 다양한 task set에 적용하기에는 어려움



https://www.researchgate.net/figure/Comparison-between-the-few-shot-classification-and-the-standard-supervised-learning_fig2_336424678

Background



Armadillo or Pangolin?

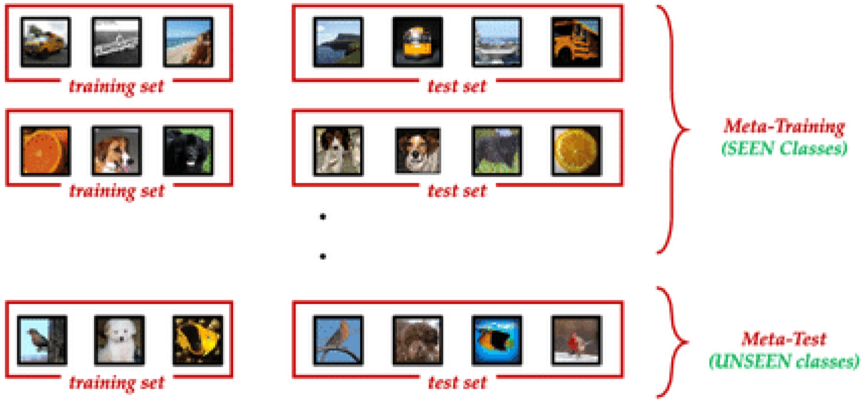
https://zzaebok.github.io/machine_learning/FSL/

Background

https://zzaebok.github.io/machine_learning/FSL/

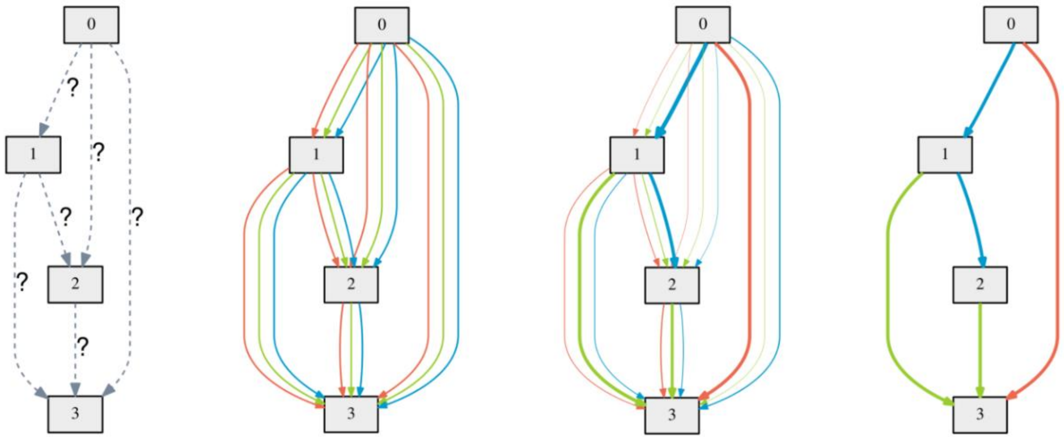
2. Related Work

Few-shot Learning via Meta-Learning



Meta-Learning for Few-Shot Classification

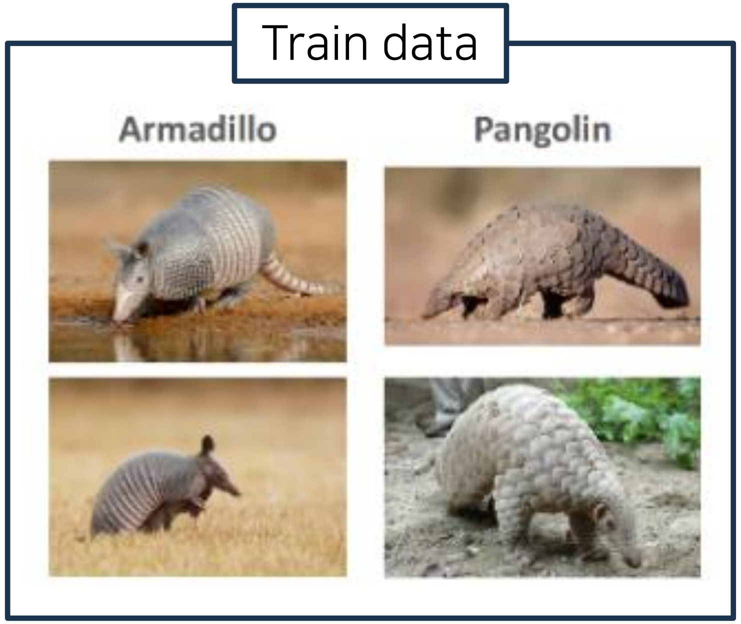
Neural Architecture Search(DARTS)

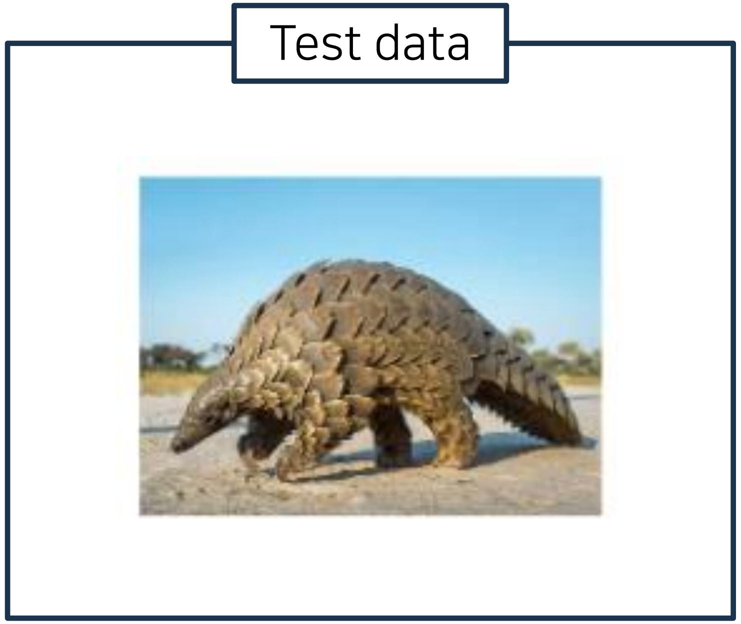


https://www.researchgate.net/figure/Comparison-between-the-few-shot-classification-and-the-standard-supervised-learning_fig2_336424678
Hanxiao Liu, Karen Simonyan, Yiming Yang. (2018). DARTS: Differentiable Architecture Search. arXiv preprint arXiv:1806.09055

Few-shot Learning via Meta-Learning

- Few-shot Learning: 몇 개의 학습 예제만으로 task를 해결하는 방법을 학습하는 문제
  - 몇 개의 학습 예제의 정의: N-way k-shot





» 2-way 2-shot classification

- 본 논문에서는 Model agnostic meta-learning(MAML) 접근 방식 사용

https://zzaebok.github.io/machine_learning/FSL/

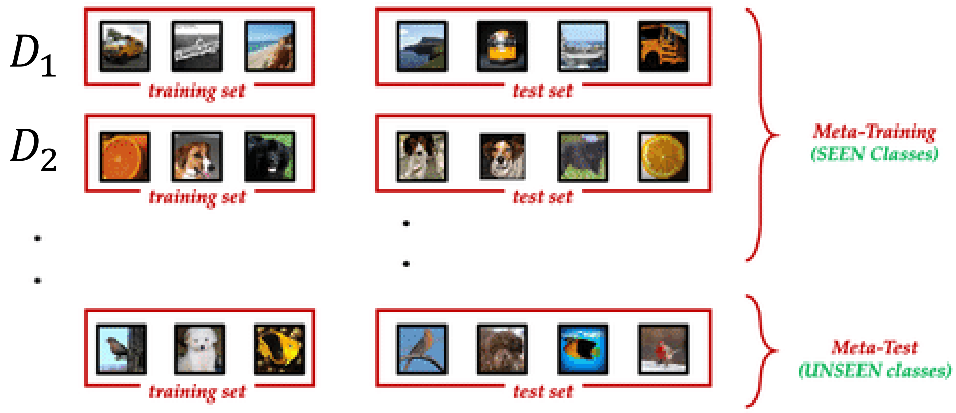
ICML(2017), 24년 6월 기준 12395회 인용

Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks

Chelsea Finn [1] **Pieter Abbeel** [1 2] **Sergey Levine** [1]

Chelsea Finn, Pieter Abbeel, Sergey Levine. (2017). Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. arXiv preprint arXiv:1703.03400

Model-Agnostic Meta-Learning



Meta-Learning for Few-Shot Classification

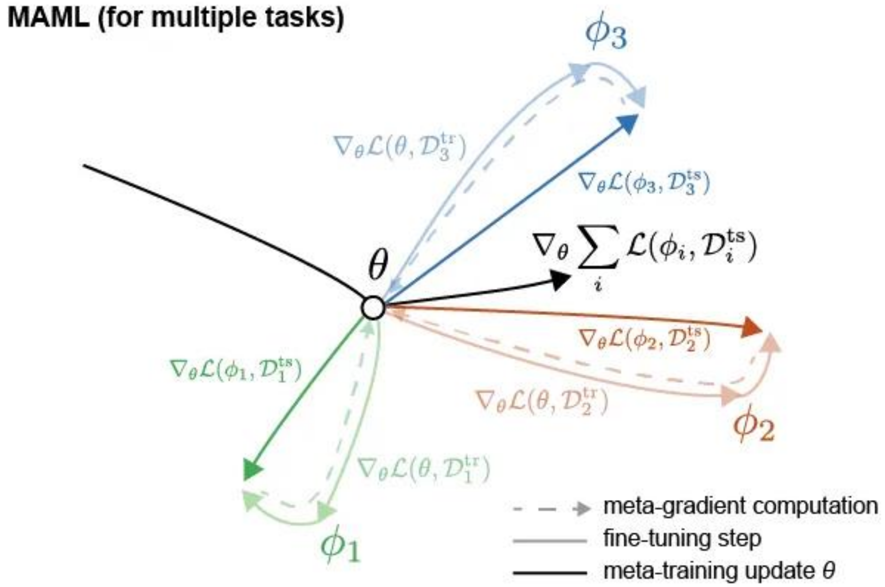
- task별 최적의 파라미터가 모두 다름
- task별 파라미터들을 바로 학습하는 것은 의미가 없음
- 새로운 task에 대해서 효율적으로 파라미터를 업데이트할 수 있는 방법을 배워야 함

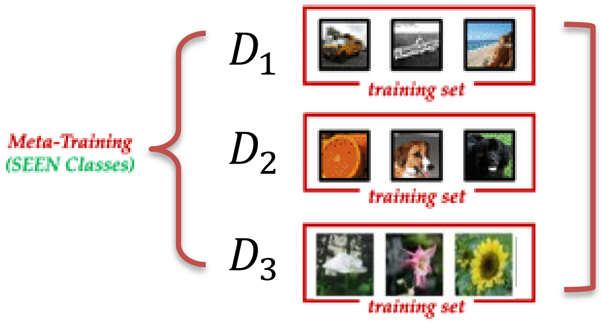
https://www.researchgate.net/figure/Comparison-between-the-few-shot-classification-and-the-standard-supervised-learning_fig2_336424678

2. Related Work

Step 1

– θ와 D_i^{tr}을 이용하여 $\emptyset_i$ 구하기





$$\emptyset_1 \leftarrow \theta - \alpha \nabla_\theta L(\theta, D_1^{tr})$$

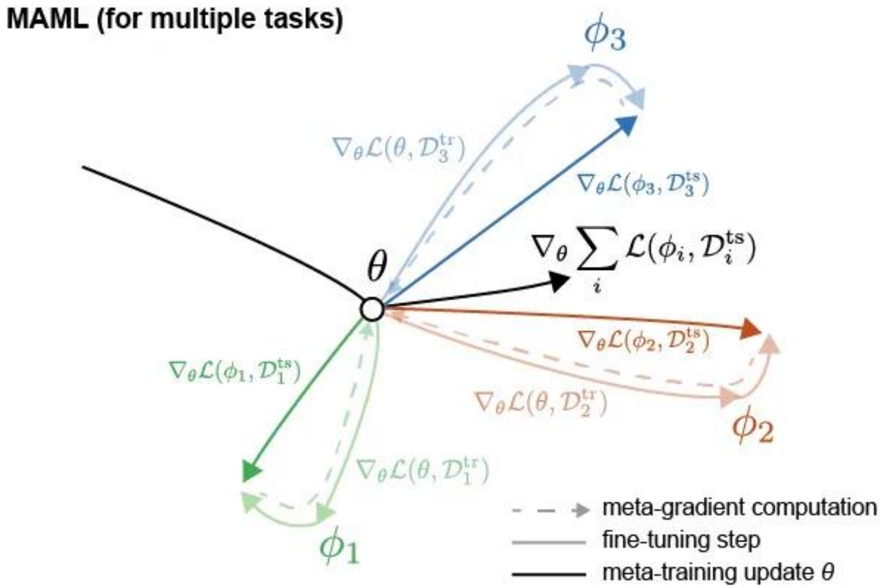
$$\emptyset_2 \leftarrow \theta - \alpha \nabla_\theta L(\theta, D_2^{tr})$$

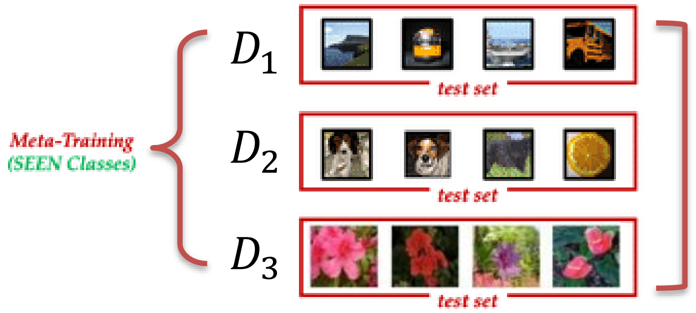
$$\emptyset_3 \leftarrow \theta - \alpha \nabla_\theta L(\theta, D_3^{tr})$$

https://www.researchgate.net/figure/Comparison-between-the-few-shot-classification-and-the-standard-supervised-learning_fig2_336424678
https://towardsdatascience.com/basics-of-few-shot-learning-with-optimization-based-meta-learning-e6e9ffd4775a

Step 2

- $\emptyset_i$와 D_i^{ts}를 이용하여 θ update

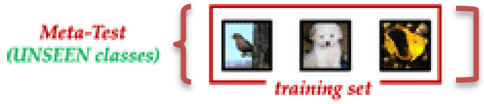




$$\theta \leftarrow \theta - \beta \nabla_\theta \sum L(\emptyset_i, D_i^{ts})$$

Step 3

- 구한 θ는 적은 학습 단계만으로 새로운 task에 대해 빠르게 적응



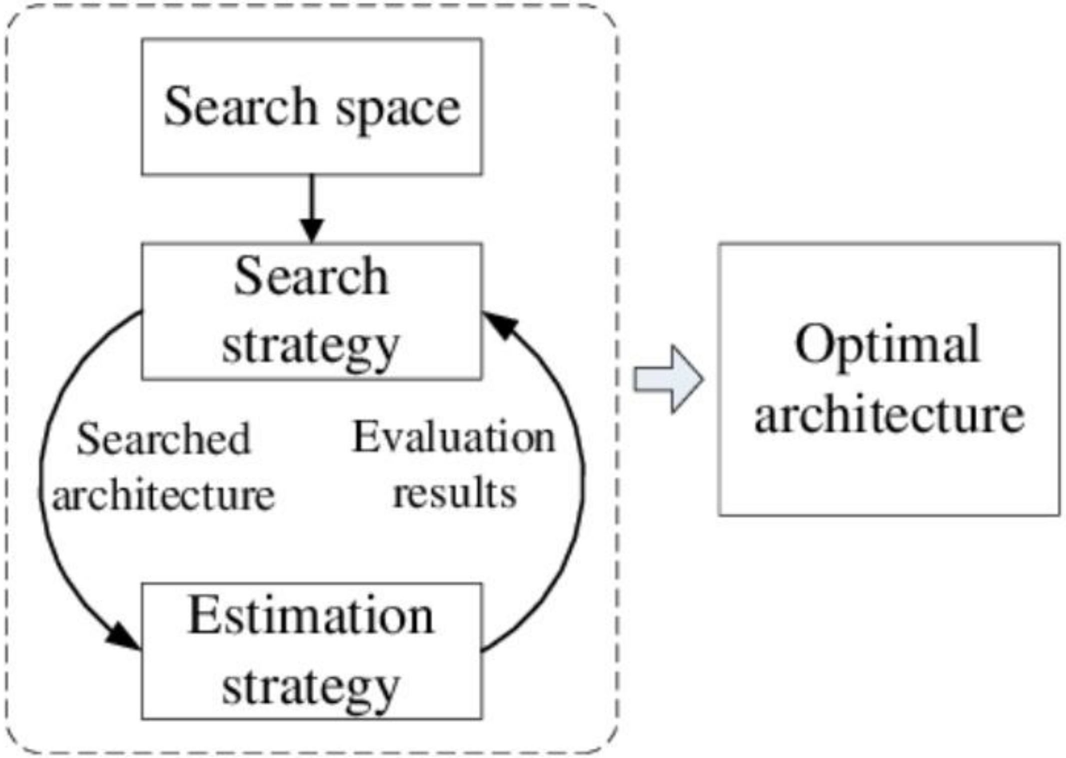
$$\theta \leftarrow \theta - \alpha \nabla_\theta \sum L(\theta, D^{test})$$

https://www.researchgate.net/figure/Comparison-between-the-few-shot-classification-and-the-standard-supervised-learning_fig2_336424678
https://towardsdatascience.com/basics-of-few-shot-learning-with-optimization-based-meta-learning-e6e9ffd4775a

2. Related Work

Neural Architecture Search

– neural architecture를 자동으로 설계하는 프로세스



https://www.researchgate.net/figure/General-process-of-neural-architecture-search_fig1_339374350

2. Related Work

ICLR(2018), 24년 6월 기준 4816회 인용

DARTS: Differentiable Architecture Search

Hanxiao Liu*
CMU
hanxiaol@cs.cmu.com

Karen Simonyan
DeepMind
simonyan@google.com

Yiming Yang
CMU
yiming@cs.cmu.edu

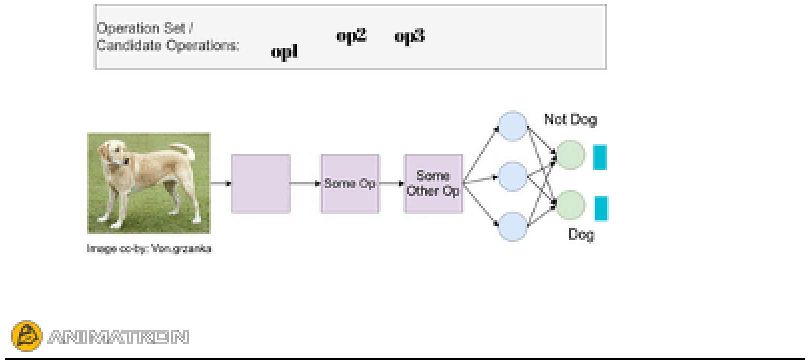
Abstract

This paper addresses the scalability challenge of architecture search by formulating the task in a differentiable manner. Unlike conventional approaches of applying evolution or reinforcement learning over a discrete and non-differentiable search space, our method is based on the continuous relaxation of the architecture representation, allowing efficient search of the architecture using gradient descent. Extensive experiments on CIFAR-10, ImageNet, Penn Treebank and WikiText-2 show that our algorithm excels in discovering high-performance convolutional architectures for image classification and recurrent architectures for language modeling, while being orders of magnitude faster than state-of-the-art non-differentiable techniques. Our implementation has been made publicly available to facilitate further research on efficient architecture search algorithms.

Hanxiao Liu, Karen Simonyan, Yiming Yang. (2018). DARTS: Differentiable Architecture Search. arXiv preprint arXiv:1806.09055

DARTS

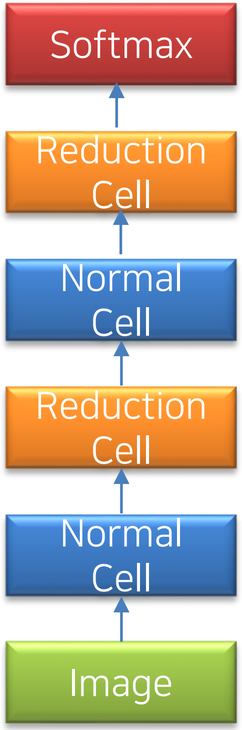
– Discrete search space?



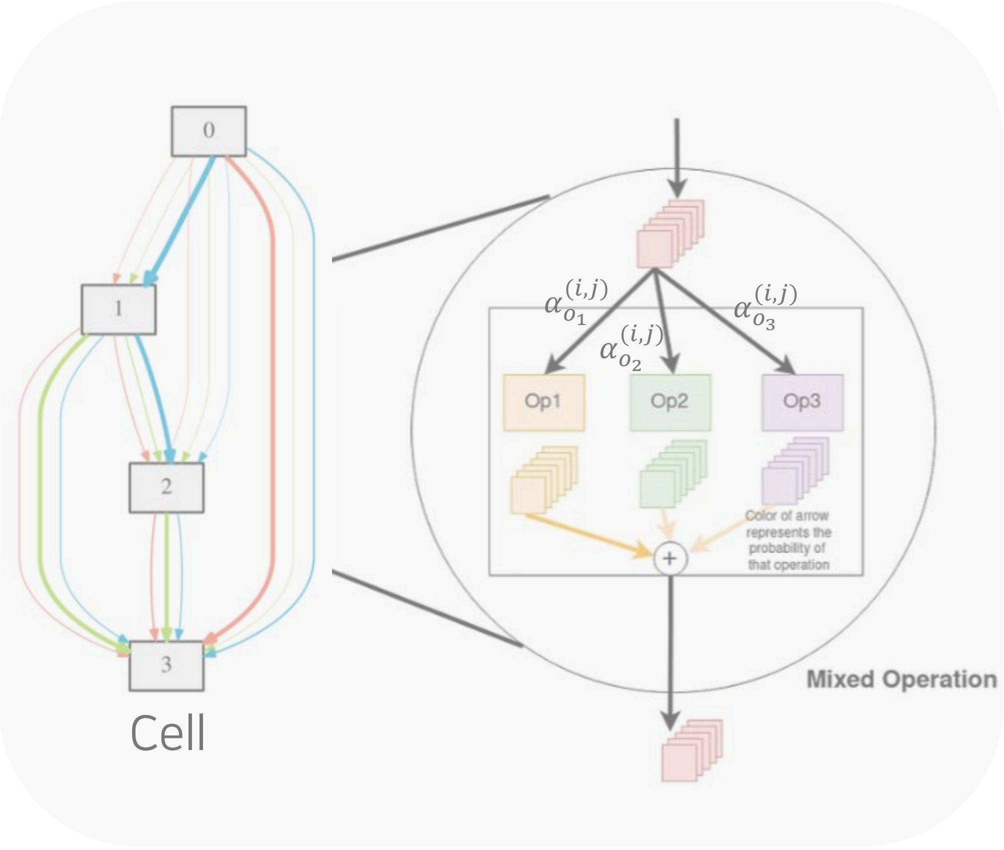
https://towardsdatascience.com/intuitive-explanation-of-differentiable-architecture-search-darts-692bdadcc69c

DARTS

– Continuous search space



Network



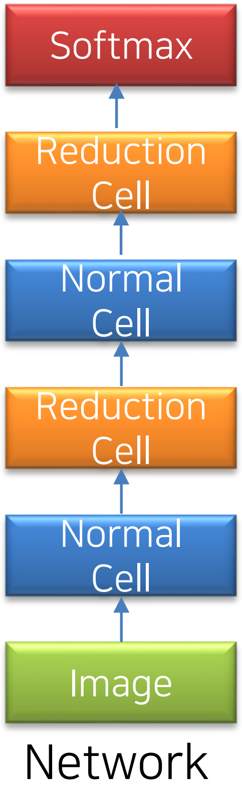
OPERATIONS =
[pooling,
standard convolution,
skip connection,
...
]

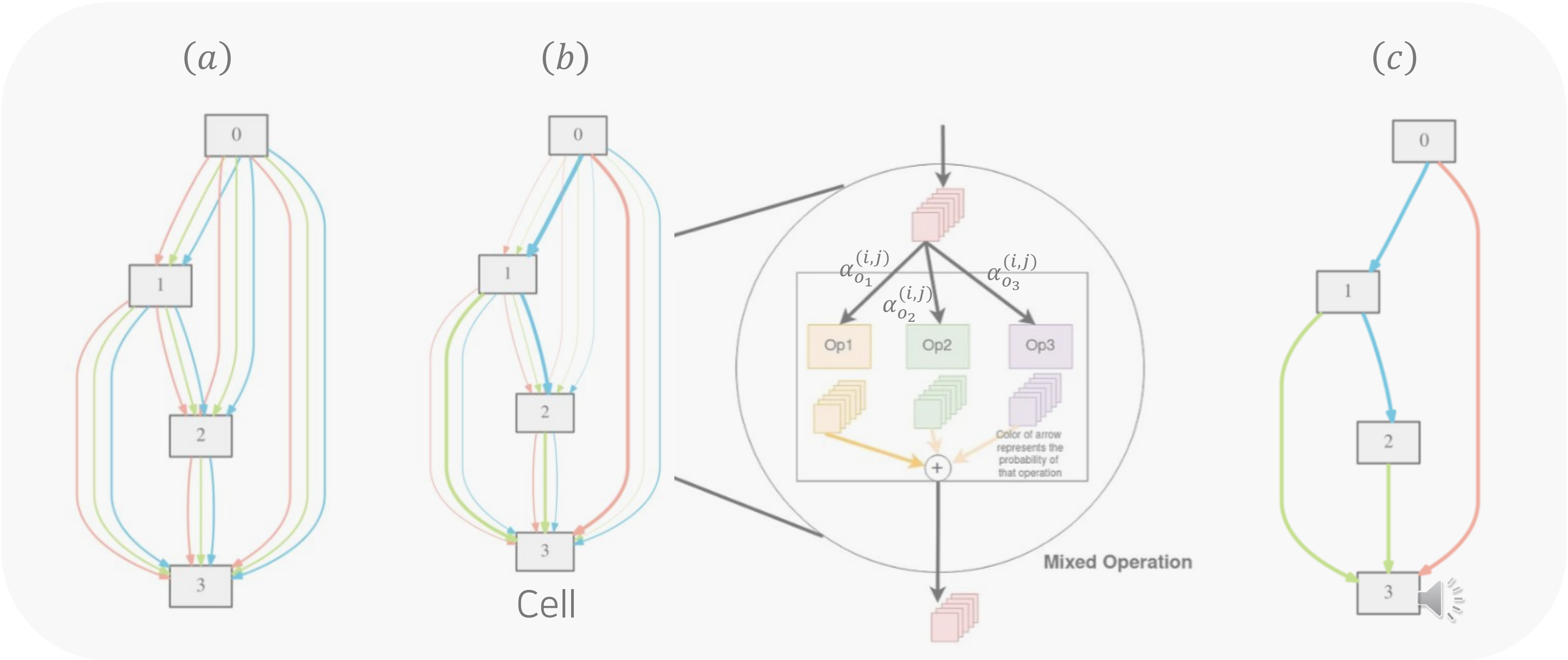
$$\bar{o}^{(i,j)}(x) = \sum_{o \in O} \frac{\exp\left(\alpha_o^{(i,j)}\right)}{\sum_{o' \in O} \exp\left(\alpha_{o'}^{(i,j)}\right)} o(x)$$

Hanxiao Liu, Karen Simonyan, Yiming Yang. (2018).
DARTS: Differentiable Architecture Search. arXiv preprint arXiv:1806.09055
https://towardsdatascience.com/intuitive-explanation-of-differentiable-architecture-search-darts-692bdadcc69c

DARTS

– Continuous search space





Hanxiao Liu, Karen Simonyan, Yiming Yang. (2018).
DARTS: Differentiable Architecture Search. arXiv preprint arXiv:1806.09055
https://towardsdatascience.com/intuitive-explanation-of-differentiable-architecture-search-darts-692bdadcc69c

선행 연구

– DBLP(2018), 24년 6월 기준 128회 인용

Transfer Learning with Neural AutoML

Catherine Wong
MIT
catwong@mit.edu

Neil Houlsby
Google AI
neilhoulsby@google.com

Yifeng Lu
Google AI
yifenglu@google.com

Andrea Gesmundo
Google AI
agesmundo@google.com

Abstract

We reduce the computational cost of Neural AutoML with transfer learning. AutoML relieves human effort by automating the design of ML algorithms. Neural AutoML has become popular for the design of deep learning architectures, however, this method has a high computation cost. To address this we propose Transfer Neural AutoML that uses knowledge from prior tasks to speed up network design. We extend RL-based architecture search methods to support parallel training on multiple tasks and then transfer the search strategy to new tasks. On language and image classification tasks, Transfer Neural AutoML reduces convergence time over single-task training by over an order of magnitude on many tasks.

– NeurIPS(2018), 24년 6월 기준 49회 인용

Auto-Meta: Automated Gradient Based Meta Learner Search

Jaehong Kim[1] **Sangyeul Lee[1]** **Sungwan Kim[1]** **Moonsu Cha[1]** **Jung Kwon Lee[1]**

Youngduck Choi[1,2] **Yongseok Choi[1]** **Dong-Yeon Cho[1]** **Jiwon Kim[1]**

SK T-Brain[1]
Yale University[2]
{ xhark, sylee0335, sw0726.kim, ckanstnzja, jklee,
yschoi, dycho24, jk} @sktbrain.com
youngduck.choi@yale.edu

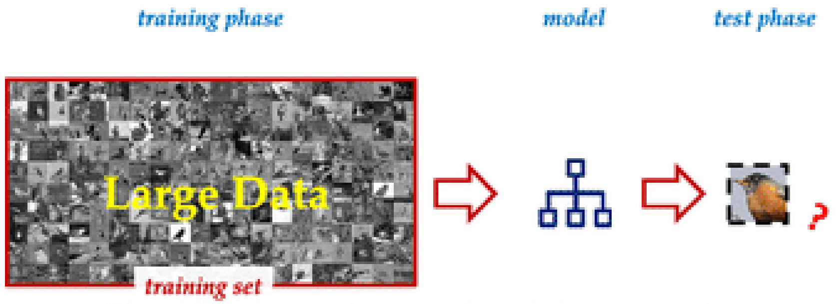
Abstract

Fully automating machine learning pipelines is one of the key challenges of current artificial intelligence research, since practical machine learning often requires costly and time-consuming human-powered processes such as model design, algorithm development, and hyperparameter tuning. In this paper, we verify that automated architecture search synergizes with the effect of gradient-based meta learning. We adopt the progressive neural architecture search [14] to find optimal architectures for meta-learners. The gradient based meta-learner whose architecture was automatically found achieved state-of-the-art results on the 5-shot 5-way Mini-ImageNet classification problem with 74.65% accuracy, which is 11.54% improvement over the result obtained by the first gradient-based meta-learner called MAML [8]. To our best knowledge, this work is the first successful neural architecture search implementation in the context of meta learning.

Catherine Wong, Neil Houlsby, Yifeng Lu, Andrea Gesmundo. (2018). Transfer Learning with Neural AutoML. arXiv preprint arXiv:1803.02780
Jaehong Kim, Sangyeul Lee, Sungwan Kim, Moonsu Cha, Jung Kwon Lee, Youngduck Choi, Yongseok Choi, Dong-Yeon Cho, Jiwon Kim. (2018). Auto-Meta: Automated Gradient Based Meta Learner Search. arXiv preprint arXiv: 1806.06927

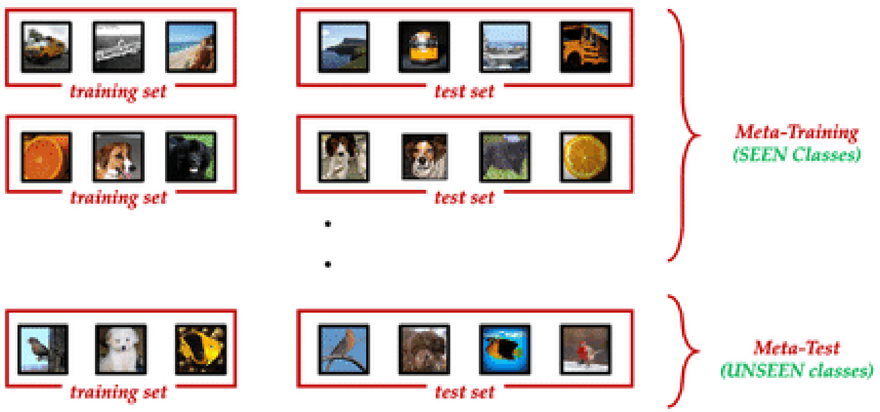
Problem Setup for Few-Shot Classification

- 고전적인 supervised deep learning



Classification with enough training examples

- Few-shot training + meta-learning



Meta-Learning for Few-Shot Classification

https://www.researchgate.net/figure/Comparison-between-the-few-shot-classification-and-the-standard-supervised-learning_fig2_336424678

Gradient-based Meta-Learning of Neural Architectures

– Meta-Objective:

$$L_{meta}\left(w, \alpha, p^{train}, \varphi^k\right) = \sum_{T_i \sim p^{train}} L_{T_i}(\Phi^k\left(w, \alpha, D_{train}^{T_i}\right), D_{test}^{T_i})$$

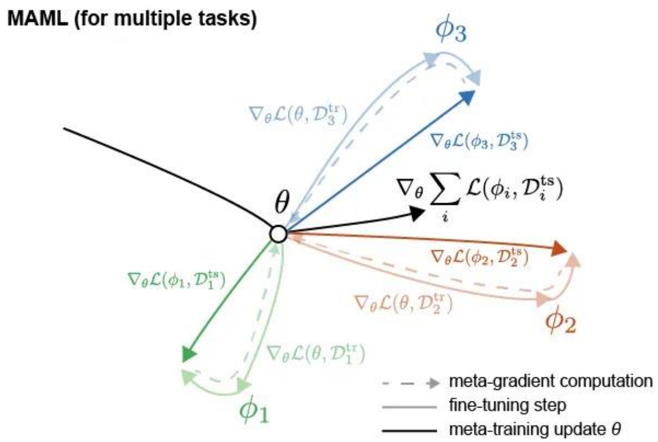
$$= \sum_{T_i \sim p^{train}} L_{T_i}(\left(w_{T_i}^*, \alpha_{T_i}^*\right), D_{test}^{T_i})$$

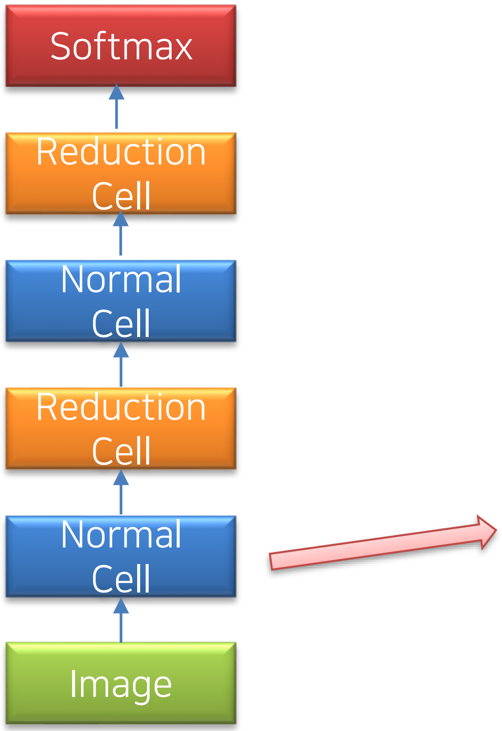
Gradient-based Meta-Learning of Neural Architectures

– Task-Learner:

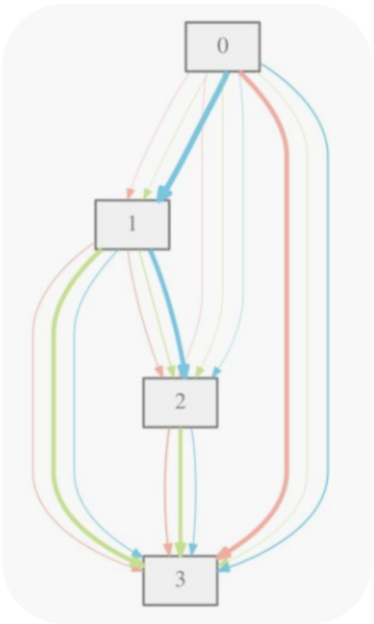
$$\begin{pmatrix} w^{j+1} \\ \alpha^{j+1} \end{pmatrix} = \Phi\left(w^j, \alpha^j, D_{train}^{T_i}\right)$$

$$= \begin{pmatrix} w^j - \lambda_{task} \nabla_w L_T(w^j, \alpha^j, D_{train}^{T_i}) \\ \alpha^j - \xi_{task} \nabla_\alpha L_T(w^j, \alpha^j, D_{train}^{T_i}) \end{pmatrix}$$





Network (parameter w)



Cell (parameter α)

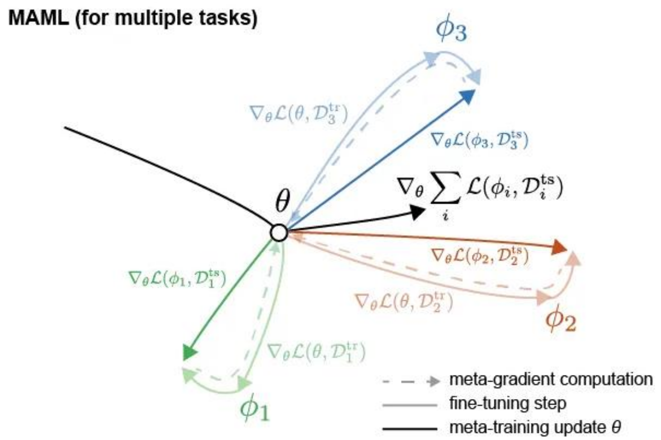
Hanxiao Liu, Karen Simonyan, Yiming Yang. (2018). DARTS: Differentiable Architecture Search. arXiv preprint arXiv:1806.09055
https://towardsdatascience.com/basics-of-few-shot-learning-with-optimization-based-meta-learning-e6e9ffd4775a

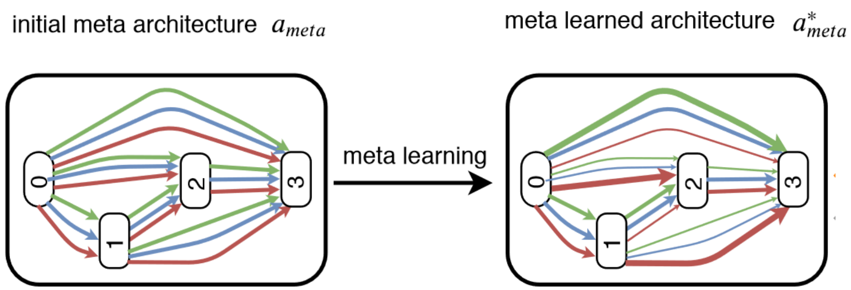
Gradient-based Meta-Learning of Neural Architectures

- Meta-Objective:

$$L_{meta}\left(w, \alpha, p^{train}, \varphi^k\right) = \sum_{T_i \sim p^{train}} L_{T_i}(\Phi^k\left(w, \alpha, D_{train}^{T_i}\right), D_{test}^{T_i})$$

$$= \sum_{T_i \sim p^{train}} L_{T_i}(\left(w_{T_i}^*, \alpha_{T_i}^*\right), D_{test}^{T_i})$$





https://towardsdatascience.com/basics-of-few-shot-learning-with-optimization-based-meta-learning-e6e9ffd4775a
Thomas Elsken, Benedikt Staffler, Jan Hendrik Metzen, Frank Hutter. (2019). Meta-Learning of Neural Architectures for Few-Shot Learning. arXiv preprint arXiv:1911.11090

Gradient-based Meta-Learning of Neural Architectures

– NAS 알고리즘을 메타 학습 알고리즘에 통합함으로써 메타 학습 알고리즘을 한 번 실행하는 것만으로 아키텍처를 탐색할 수 있음

Algorithm 1 METANAS: Meta-Learning of Neural Architectures

Input:
distribution over tasks $p(\mathcal{T})$,
task-learner $\Phi^k(w, a, D_{train}^{\mathcal{T}_i})$ # e.g. DARTS [32]
meta-learner Ψ_w, Ψ_α # e.g. REPTILE [36]
Initialize w_{meta}, α_{meta}
while not converged **do**
  Sample tasks $\mathcal{T}_1, \ldots, \mathcal{T}_n$ from $p(\mathcal{T})$
  for all $\mathcal{T}_i$ **do**
    $w_{\mathcal{T}_i}^*, \alpha_{\mathcal{T}_i}^* \leftarrow \Phi^k(w_{meta}, \alpha_{meta}, D_{train}^{\mathcal{T}_i})$
  end for
  $w_{meta} \leftarrow \Psi_w\left(w_{meta}, \{w_{\mathcal{T}_i}^*, \alpha_{\mathcal{T}_i}^*, \mathcal{T}_i\}_{i=1}^n\right)$
  $\alpha_{meta} \leftarrow \Psi_\alpha\left(\alpha_{meta}, \{w_{\mathcal{T}_i}^*, \alpha_{\mathcal{T}_i}^*, \mathcal{T}_i\}_{i=1}^n\right)$
end while
return w_{meta}, α_{meta}

Task-Learning(DARTS)

Meta-Learning(MAML)

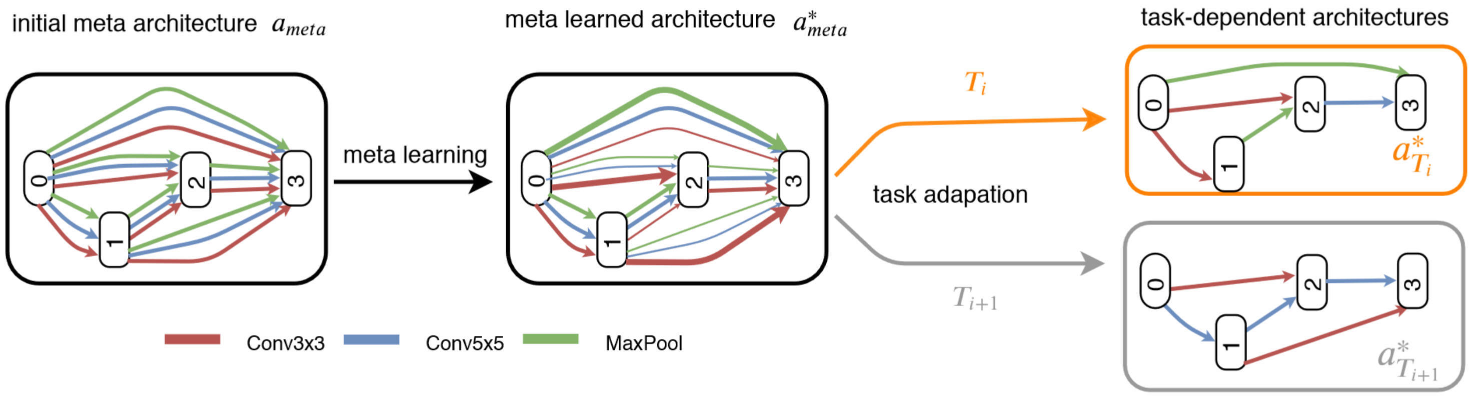
Thomas Elsken, Benedikt Staffler, Jan Hendrik Metzen, Frank Hutter. (2019). Meta-Learning of Neural Architectures for Few-Shot Learning. arXiv preprint arXiv:1911.11090

4. Task-dependent Architecture Adaptation

Learning of new task after meta-learning

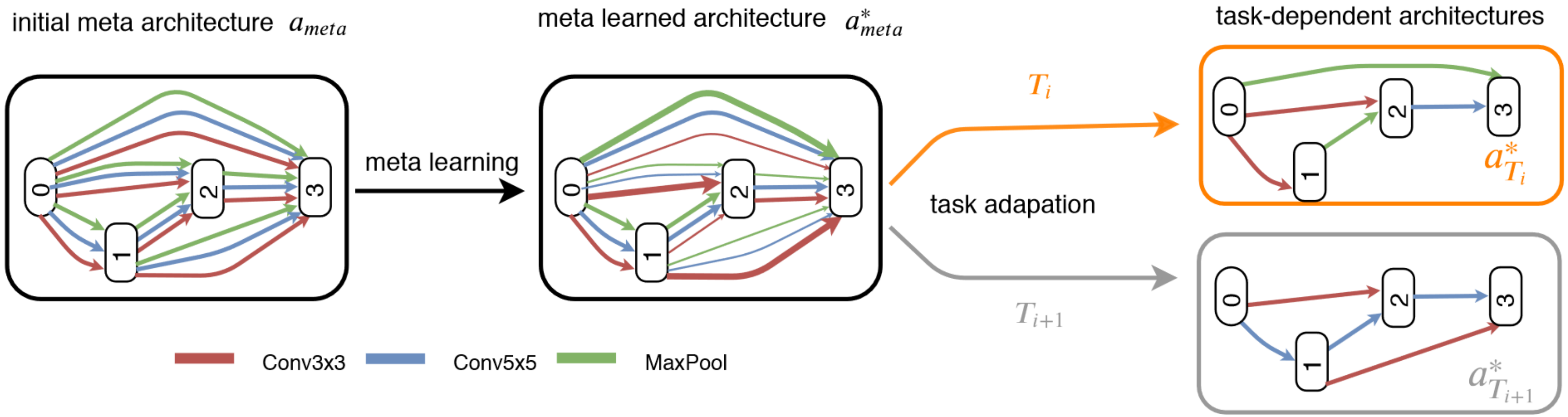
Algorithm 3 Learning of new task after meta-learning (i.e., meta-testing) with DARTS.

Input: new task $\mathcal{T} = (D_{train}, D_{test})$
meta-learned architecture and weights α_{meta}, w_{meta}
$w_{\mathcal{T}} \leftarrow w_{meta}$
$\alpha_{\mathcal{T}} \leftarrow \alpha_{meta}$
for $j \leftarrow 1, \ldots, k$ **do**
    $w_{\mathcal{T}} \leftarrow w_{\mathcal{T}} - \lambda_{task} \nabla_w \mathcal{L}_{\mathcal{T}}(w_{\mathcal{T}}, \alpha_{\mathcal{T}}, D_{train})$
    $\alpha_{\mathcal{T}} \leftarrow \alpha_{\mathcal{T}} - \xi_{task} \nabla_\alpha \mathcal{L}_{\mathcal{T}}(w_{\mathcal{T}}, \alpha_{\mathcal{T}}, D_{train})$
end for
$\bar{\alpha}_{\mathcal{T}} \leftarrow \text{PRUNE}(\alpha_{\mathcal{T}})$
Evaluate D_{test} with $\bar{\alpha}_{\mathcal{T}}, w_{\mathcal{T}}$



Thomas Elsken, Benedikt Staffler, Jan Hendrik Metzen, Frank Hutter. (2019). Meta-Learning of Neural Architectures for Few-Shot Learning. arXiv preprint arXiv:1911.11090

Learning of new task after meta-learning



- DARTS task optimizer가 탐색한 이 아키텍처의 혼합 가중치는 엄격하게 0 또는 1로 이루어지지 않음
  - 이러한 hard-pruning은 성능을 크게 저하시키므로 pruning을 거친 아키텍처는 재학습이 필요
    - 본 연구에서는 soft-pruning으로 보완

Thomas Elsken, Benedikt Staffler, Jan Hendrik Metzen, Frank Hutter. (2019). Meta-Learning of Neural Architectures for Few-Shot Learning. arXiv preprint arXiv:1911.11090

Soft-Pruning of Mixture over Operations

– 혼합 연산을 구성하는 연산들의 혼합 가중치 sparsify

$$\hat{\alpha}_o^{i,j} = \frac{\exp\left(\alpha_o^{i,j}\right)}{\sum_{o' \in O} \exp\left(\alpha_{o'}^{i,j}\right)}$$

$$\hat{\alpha}_{\tau_\alpha}^{i,j}(o) = \frac{\exp\left(\alpha_o^{i,j} / \tau_\alpha\right)}{\sum_{o' \in O} \exp\left(\alpha_{o'}^{i,j} / \tau_\alpha\right)}$$

- Task training 과정에서 0으로 annealing 되는 temperature τ_α 추가

Soft-Pruning of Mixture over Input Nodes

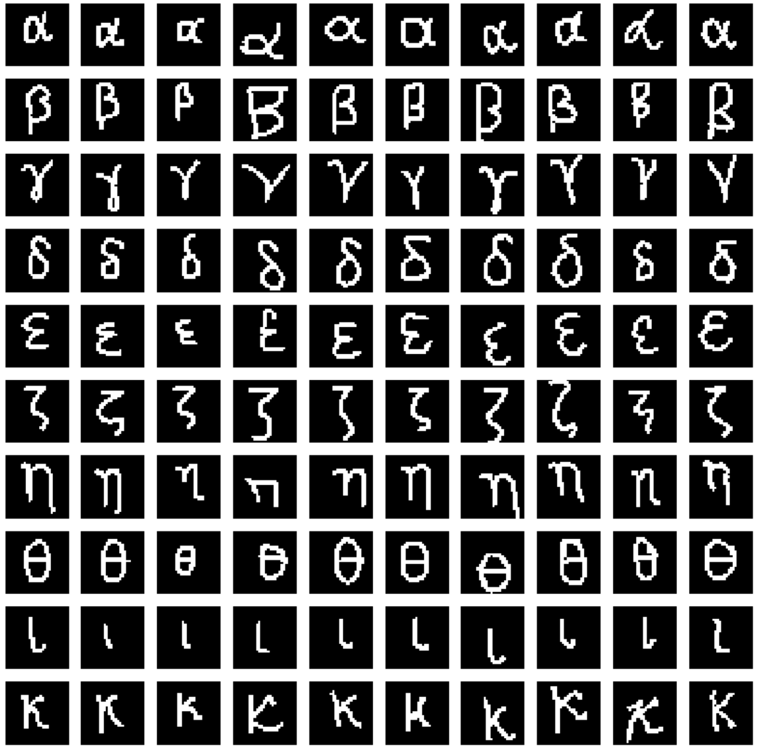
– 입력의 가중치를 operation의 가중치와 동일한 방식으로 sparsify

$$x^{(j)} = \sum_{i<j} \frac{\exp\left(\frac{\beta^{(i,j)}}{\tau_\beta}\right)}{\sum_{k<j} \exp\left(\frac{\beta^{(k,j)}}{\tau_\beta}\right)} MixedOp(x^{(i)})$$

- Task training 과정에서 0으로 annealing 되는 temperature τ_β 추가

Omniglot dataset

- 서로 다른 알파벳 1623개로 이루어짐(1623 class)
- 알파벳 별로 각각 다른 사람이 쓴 20개의 이미지 샘플 존재

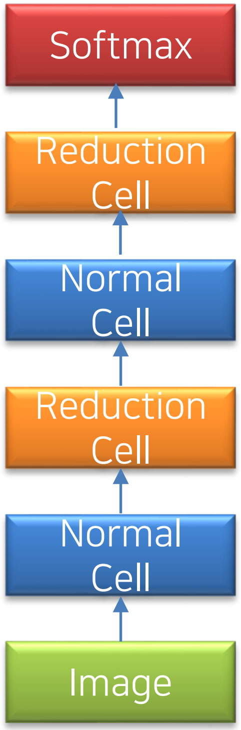


https://github.com/topics/omniglot

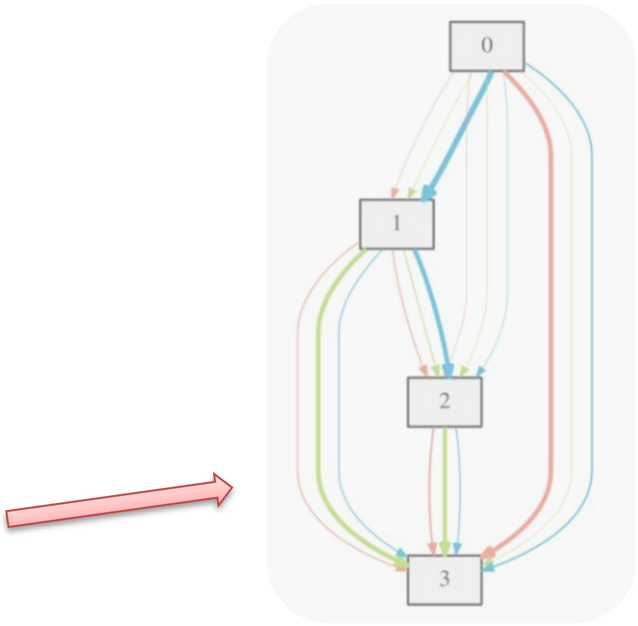
5. Experiments

DARTS

– Search Space



Network

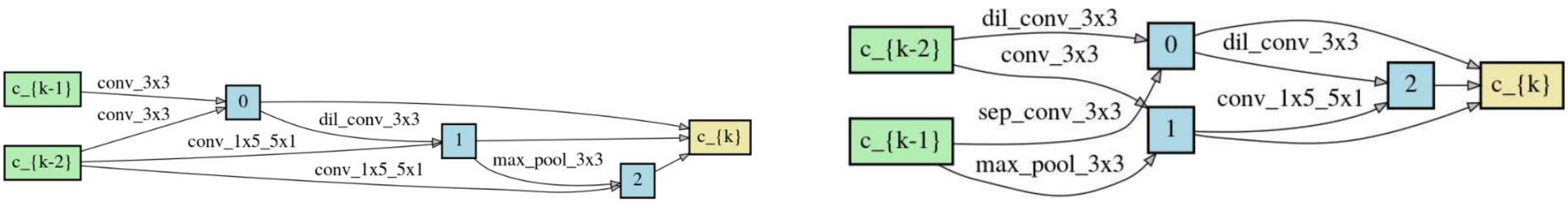


Cell

OPERATIONS =
[MaxPool 3x3,
AvgPool 3x3,
Conv 1x5-5x1,
Conv 3x3,
SepConv 3x3,
Dilated Conv 3x3,
skip connection
]

Hanxiao Liu, Karen Simonyan, Yiming Yang. (2018). DARTS: Differentiable Architecture Search. arXiv preprint arXiv:1806.09055

Results

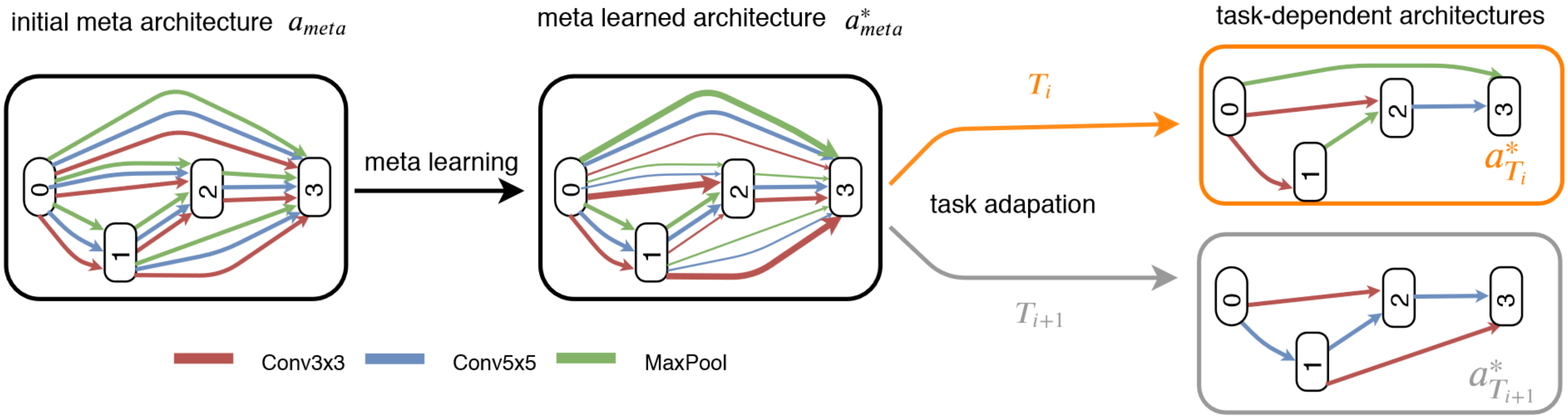


(a) Normal cell. (b) Reduction cell.

		Omniglot
Method	# params	1-shot, 20 way
MAML[16]	30k	95.8 ± 0.03
MAML++[1]	-	$\mathbf{97.65 \pm 0.05}$
T-NAS++[29]	27k	-
REPTILE[36]	30k	89.43 ± 0.14
AutoMeta[24]	100k	-
METANAS *	1M	$\mathbf{97.74 \pm 0.08}$
TADAM [37]	12M	-
MetaOptNet [27][1]	12M	-
LEO [42]	36.5M	-

Thomas Elsken, Benedikt Staffler, Jan Hendrik Metzen, Frank Hutter. (2019). Meta-Learning of Neural Architectures for Few-Shot Learning. arXiv preprint arXiv:1911.11090

6. Conclusion



Thomas Elsken, Benedikt Staffler, Jan Hendrik Metzen, Frank Hutter. (2019). Meta-Learning of Neural Architectures for Few-Shot Learning. arXiv preprint arXiv:1911.11090

감사합니다