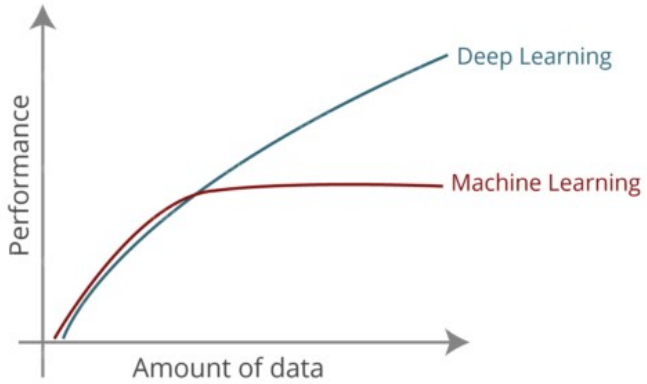
Meta Learning

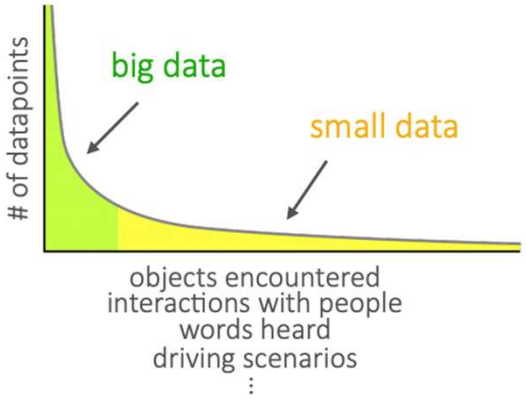
Introduction

What is Meta-Learning?

‘Meta’ is a prefix used in English to indicate a concept which is an abstraction behind another concept, used to complete or add to the latter.

‘Meta Learning’, training time동안 접하지 않았던 새로운 Task나 Environment 에 대해서 잘 적응하거나 일반화가 잘 되는 것.





데이터 의존도를 낮추기 위한 방법들,
Domain adaptation, Semi-supervised learning, Self-supervised learning, Meta learning …

Introduction

What is Meta-Learning?

학습을 위한 학습 (Learning to learn)

학습 : 데이터셋을 이용한 일반화 과정

학습이 잘 되는 모델 :

적은 데이터로도 적절한 일반화가 가능한 모델

적은 데이터로도 일반화가
가능하도록 학습시키는 것이 목표

- 고양이가 없는 이미지를 학습시킨 classifier도 몇개의 고양이 사진을 본 후에는 test image 상에 고양이가 있는지 여부를 판단할 수 있다.
- 게임 봇이 새로운 게임에 대해서 빠르게 마스터할 수 있다.
- 평평한 지면 환경에서만 학습해 온 미니 로봇이 경사진 환경에서도 task를 수행할 수 있다.

Introduction

Meta Learning의 종류

1. (Metric based) Efficient distance metric 을 학습하는 방식
   - Siamese Neural network for one-shot image recognition
   - Matching networks for one-shot learning
   - Prototypical Networks for Few-shot learning
   - Learning to Compare : Relation Network for Few-shot Learning
2. (Model based) External / internal memory 를 위한 (Recurrent) Network를 사용하는 방식
3. (Graph based) Graph Neural Network를 최적화하는 방식
   - Few-shot learning with graph neural networks
   - Transductive propagation network for few-shot learning
4. (Optimization based) Fast learning을 위한 model parameter를 최적화하는 방식
   - Model-Agnostic Meta-learning for Fast Adaptation of Deep Networks (MAML)
   - FOMAML, Reptile, MAML++, L2F-MAML 등

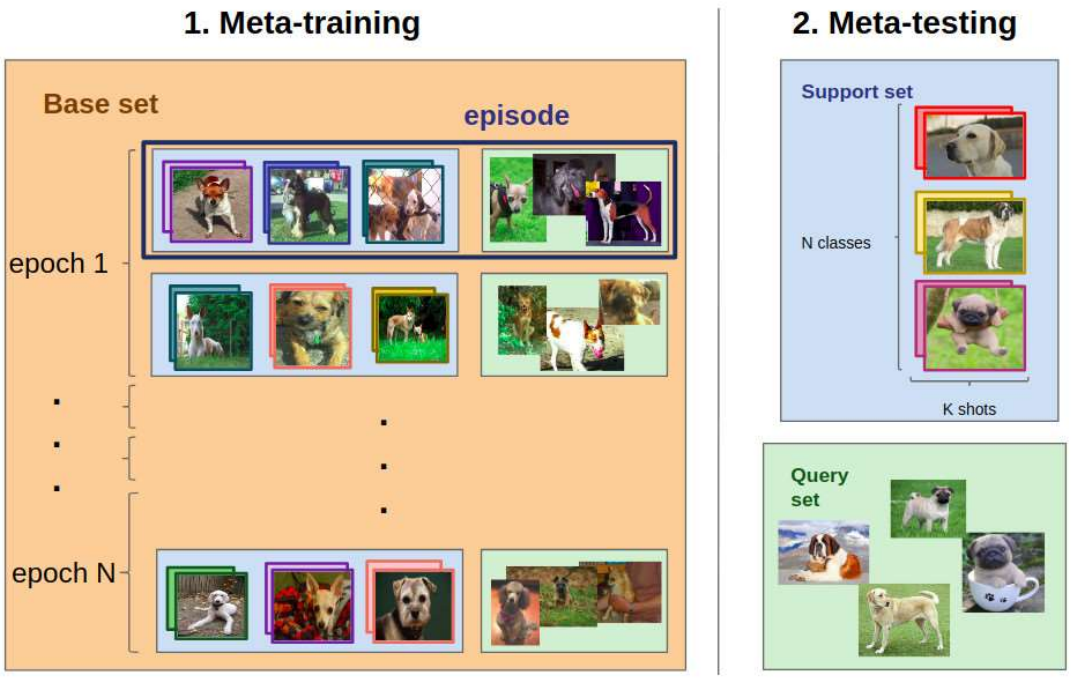
Few-shot learning

Definition

- **N-way, K-shot classification**

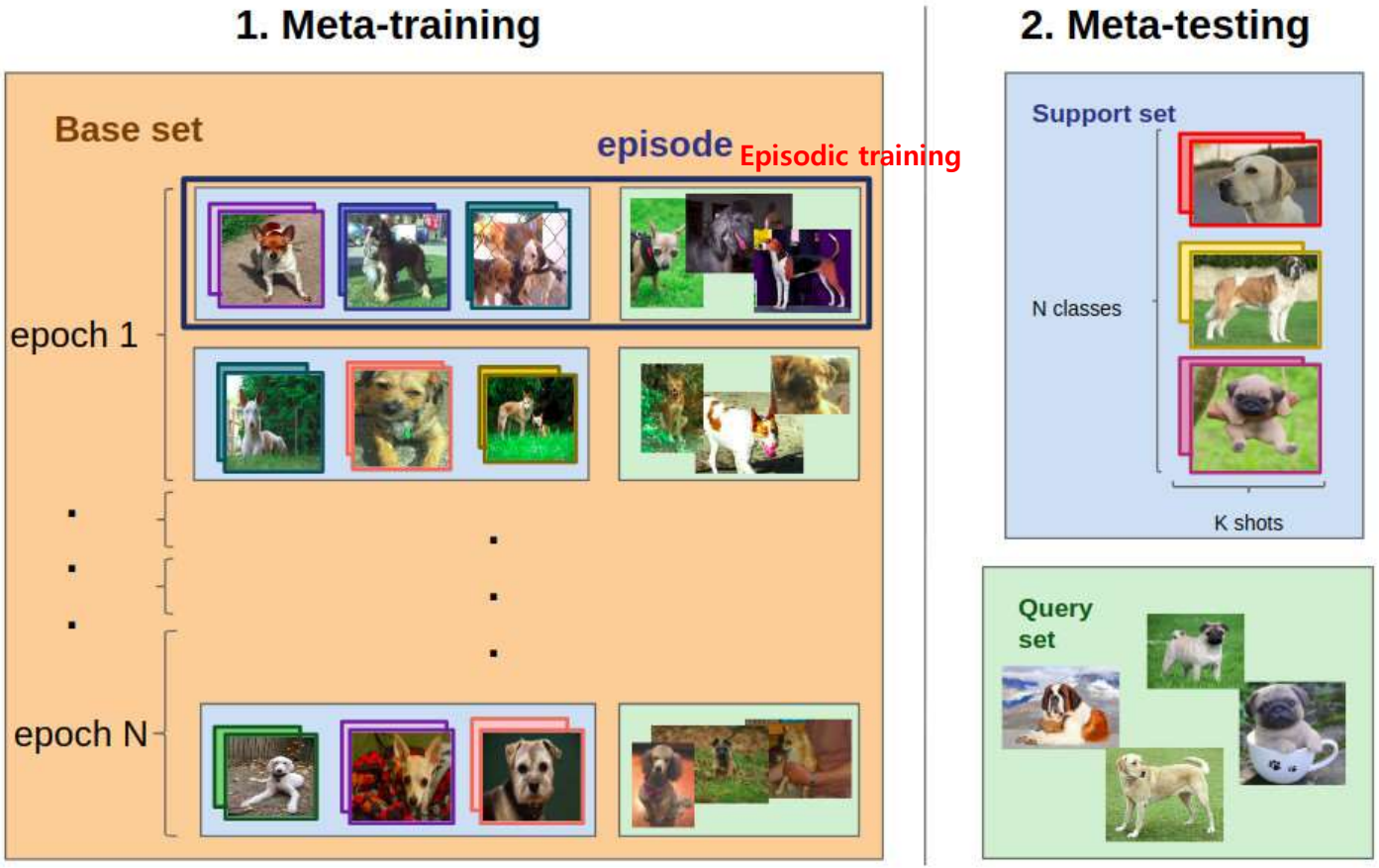
  N : class의 수, K : example 수 (실제 보는 데이터)

- **Support set**
  - 새로운 클래스에 대한 학습 데이터셋 (≒ 트레이닝셋)
- **Query set**
  - 새로운 클래스에 대한 테스트 데이터셋 (≒ 테스트셋)
- **Meta-test dataset**
  - Support set + Query set
- **Meta-train dataset**
  - Meta-learning의 dataset



https://www.kdnuggets.com/2020/03/few-shot-image-classification-meta-learning.html

Few-shot learning



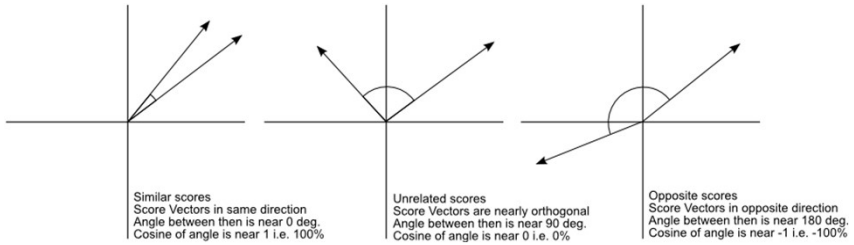
https://www.kdnuggets.com/2020/03/few-shot-image-classification-meta-learning.html

Few-shot learning

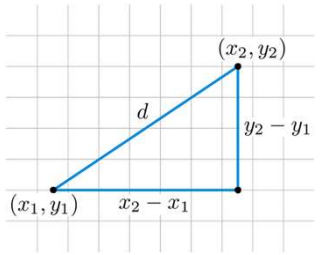
Similarity Measure

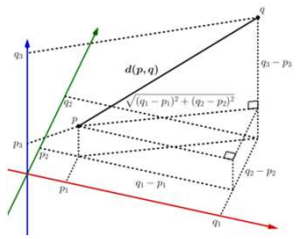
Cosine distance

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}$$



Euclidean distance





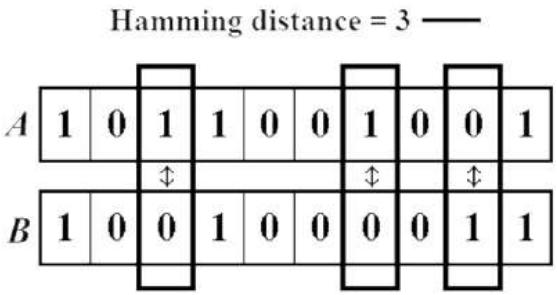
$$d(\mathbf{p}, \mathbf{q}) = d(\mathbf{q}, \mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$

$$= \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}.$$

Few-shot learning

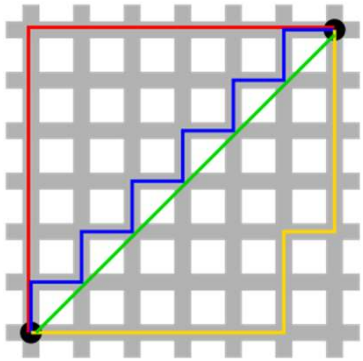
Similarity Measure

Hamming distance



- "**karolin**" and "**kathrin**" is 3.
- "**karolin**" and "**kerstin**" is 3.
- "**kathrin**" and "**kerstin**" is 4.
- **1011101** and **1001001** is 2.
- **2173896** and **2233796** is 3.

Manhattan distance



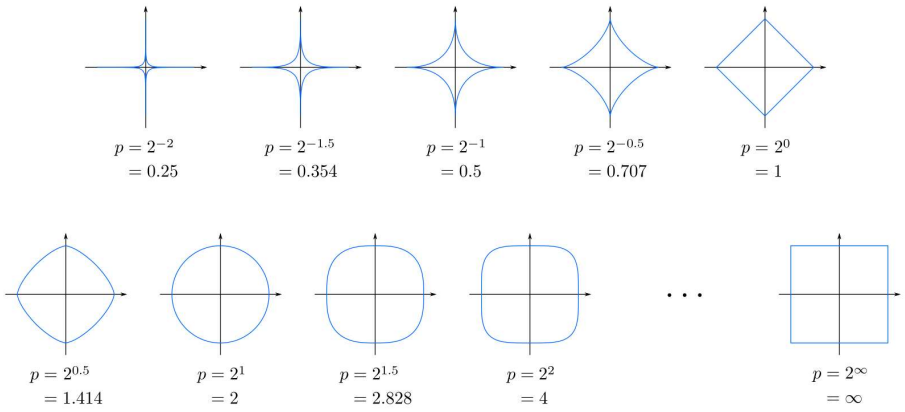
$$d_1(\mathbf{p}, \mathbf{q}) = \|\mathbf{p} - \mathbf{q}\|_1 = \sum_{i=1}^{n} |p_i - q_i|$$

Few-shot learning

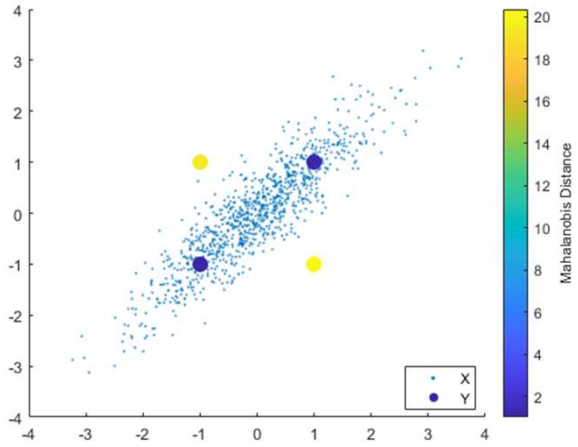
Similarity Measure

Minkowski distance

$$D(X,Y) = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p}$$



Mahalanobis distance



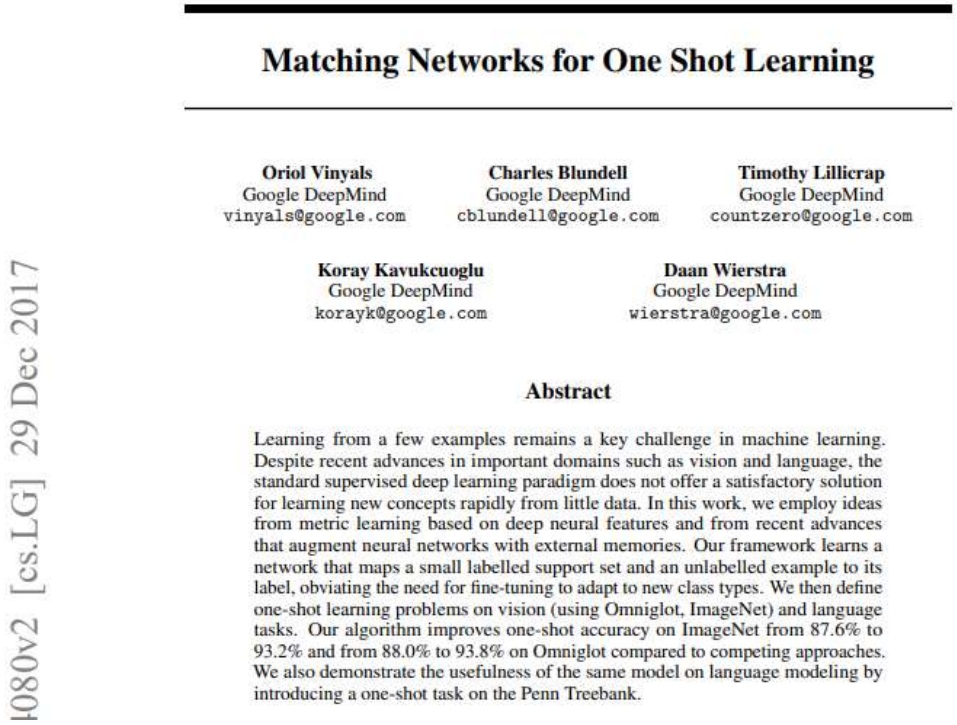
$$d_M(X_q, X_i) = \sqrt{(X_q - X_i)^T S^{-1} (X_q - X_i)}$$

where S is the covariance matrix

Metric based learning

Matching Networks for One shot Learning

- 모델의 모수적인(Parametric) 성질때문에 학습이 느리다.
- 비모수적 방법을 사용하여 모델을 빨리 학습 (k-nn 등)
- Common example에 대한 일반화 성능(모수적 방법) 을 유지하면서 새로운 데이터에 대해서는 빠른 습득을 할 수 있는 비모수적 방법 제안
- Episodic tasks learning : test / train condition 동일하게



Matching Networks for One Shot Learning

Oriol Vinyals
Google DeepMind
vinyals@google.com

Charles Blundell
Google DeepMind
cblundell@google.com

Timothy Lillicrap
Google DeepMind
countzero@google.com

Koray Kavukcuoglu
Google DeepMind
korayk@google.com

Daan Wierstra
Google DeepMind
wierstra@google.com

Abstract

Learning from a few examples remains a key challenge in machine learning. Despite recent advances in important domains such as vision and language, the standard supervised deep learning paradigm does not offer a satisfactory solution for learning new concepts rapidly from little data. In this work, we employ ideas from metric learning based on deep neural features and from recent advances that augment neural networks with external memories. Our framework learns a network that maps a small labelled support set and an unlabelled example to its label, obviating the need for fine-tuning to adapt to new class types. We then define one-shot learning problems on vision (using Omniglot, ImageNet) and language tasks. Our algorithm improves one-shot accuracy on ImageNet from 87.6% to 93.2% and from 88.0% to 93.8% on Omniglot compared to competing approaches. We also demonstrate the usefulness of the same model on language modeling by introducing a one-shot task on the Penn Treebank.

Metric based learning

Matching Networks for One shot Learning

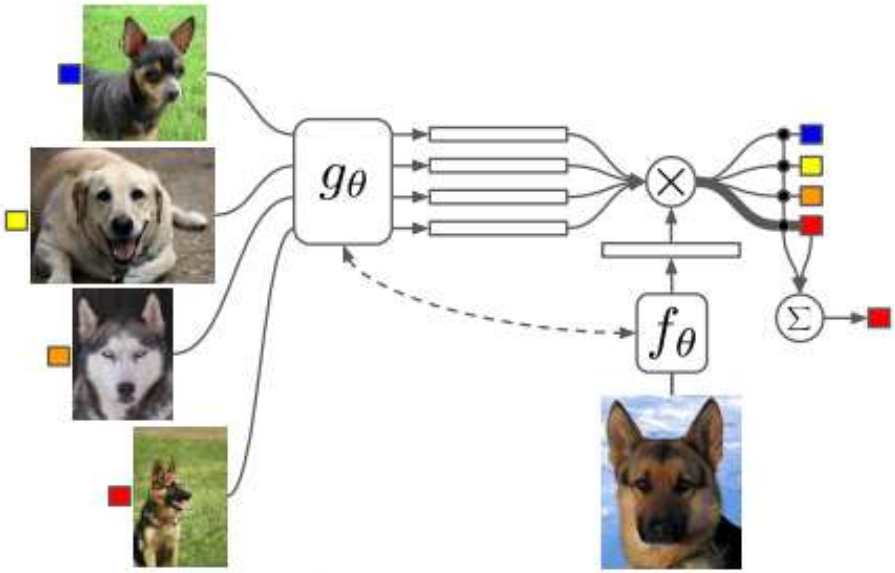
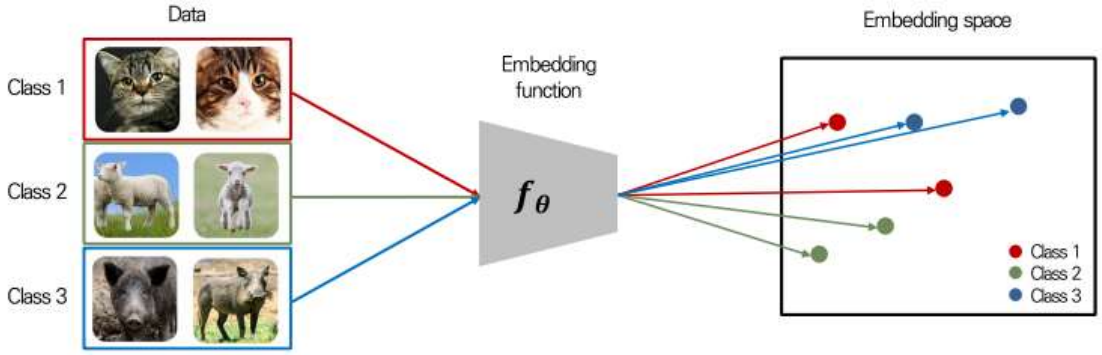


Figure 1: Matching Networks architecture

- **Embedding function**
  - g_θ : Support set의 임베딩 함수
  - f_θ : Query set의 임베딩 함수



Metric based learning

Matching Networks for One shot Learning

- **Distance**
  - Cosine similarity (attention) $\hat{y} = \sum_{i=1}^{k} a(\hat{x}, x_i) y_i$
  - $\boldsymbol{a(x, x_i) = \frac{exp(cosine(f(x), g(x_i)))}{\sum_{j=1}^{k} exp(cosine(f(x), g(x_j)))}}$

- **Training strategy**

$$\theta = \arg\max_{\theta} E_{L \sim T} \left[E_{S \sim L, B \sim L} \left[\sum_{(x,y) \in B} \log P_\theta (y | x, S) \right] \right].$$

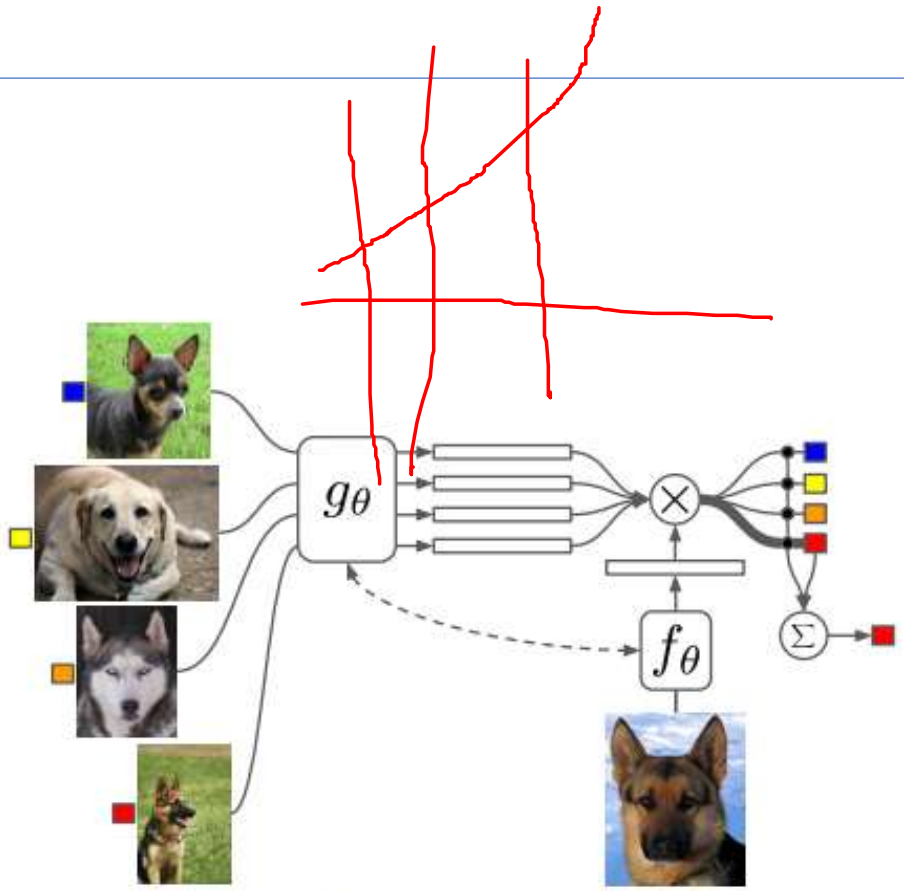
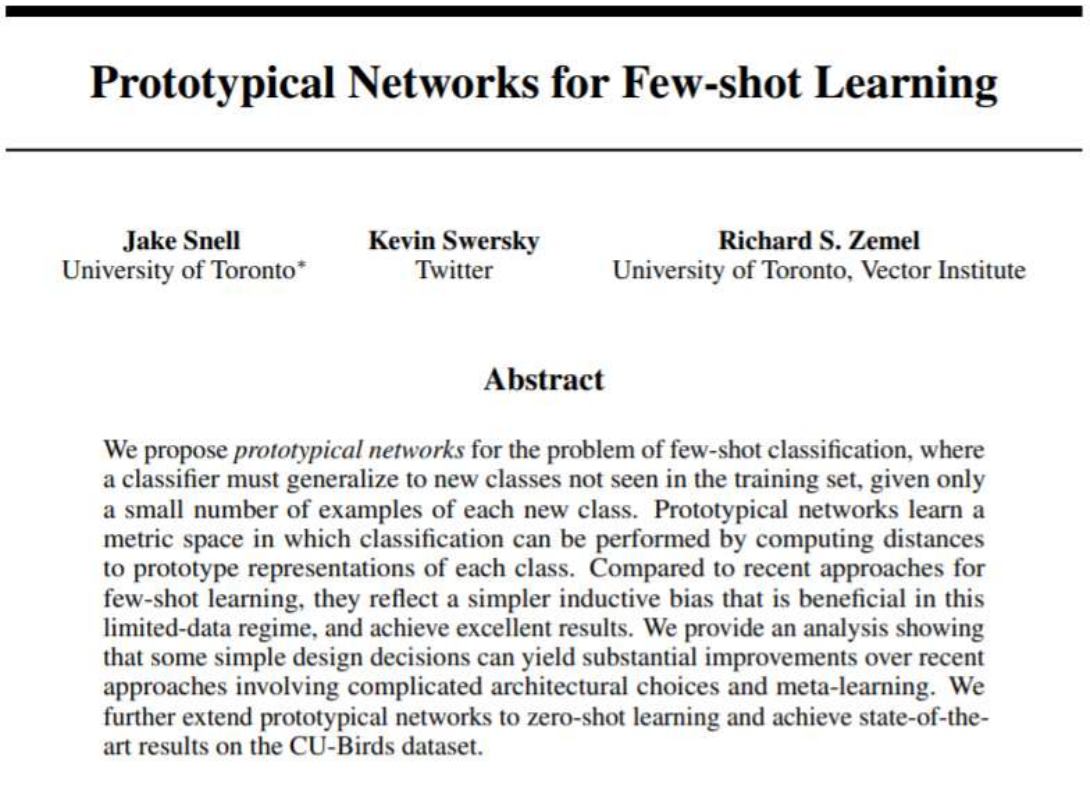


Figure 1: Matching Networks architecture

Metric based learning

Prototypical Networks for Few-shot Learning

- 범주별 서포트 데이터의 평균 위치인 프로토타입(Prototype) 이라는 개념 사용 → 프로토타입 벡터와 쿼리 벡터간의 거리 계산
- 쿼리 예측에 필요한 계산량을 N * K 에서 N개로 줄일 수 있음



Prototypical Networks for Few-shot Learning

Jake Snell
University of Toronto*

Kevin Swersky
Twitter

Richard S. Zemel
University of Toronto, Vector Institute

Abstract

We propose *prototypical networks* for the problem of few-shot classification, where a classifier must generalize to new classes not seen in the training set, given only a small number of examples of each new class. Prototypical networks learn a metric space in which classification can be performed by computing distances to prototype representations of each class. Compared to recent approaches for few-shot learning, they reflect a simpler inductive bias that is beneficial in this limited-data regime, and achieve excellent results. We provide an analysis showing that some simple design decisions can yield substantial improvements over recent approaches involving complicated architectural choices and meta-learning. We further extend prototypical networks to zero-shot learning and achieve state-of-the-art results on the CU-Birds dataset.

Metric based learning

Prototypical Networks for Few-shot Learning

- **Embedding function**
  - f_θ : Query set의 임베딩 함수
- **Prototype**
  - $$\mathbf{c}_k = \frac{1}{|S_k|} \sum_{(\mathbf{x}_i, y_i) \in S_k} f_\phi(\mathbf{x}_i)$$
  - 모든 Support embedding vector와 거리 계산하지 않아도 됨
- **Distance**
  - Euclidean distance 사용

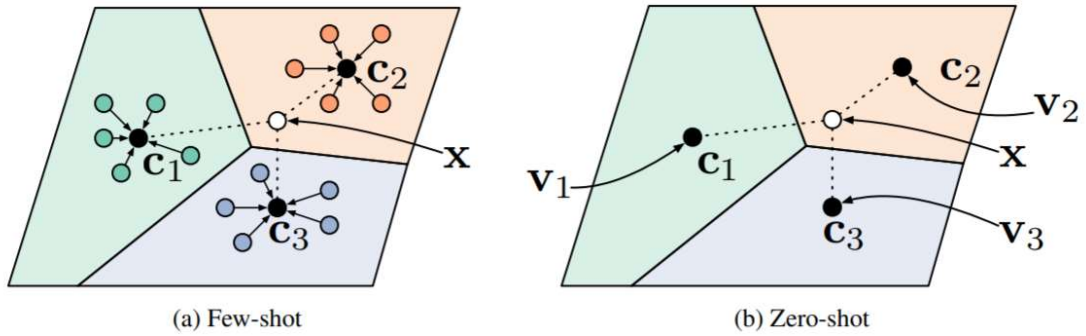
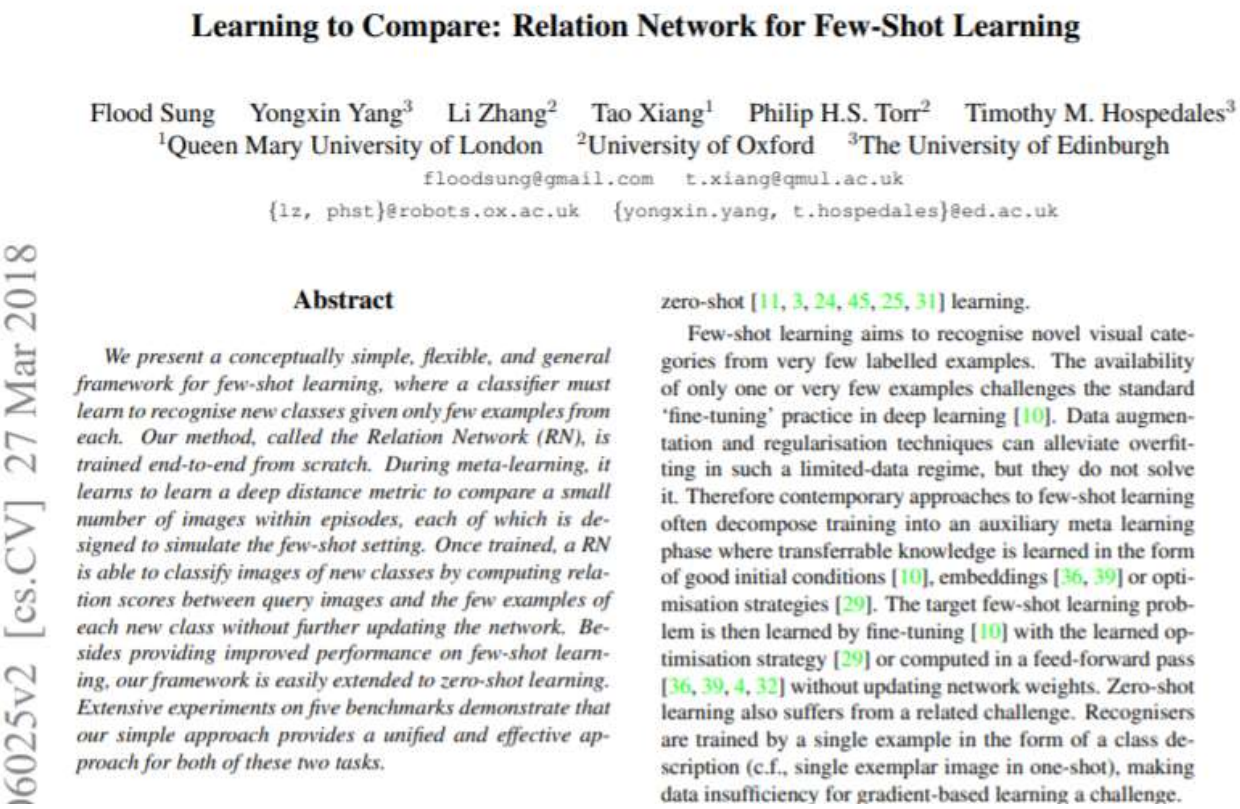


Figure 1: Prototypical networks in the few-shot and zero-shot scenarios. **Left**: Few-shot prototypes $\mathbf{c}_k$ are computed as the mean of embedded support examples for each class. **Right**: Zero-shot prototypes $\mathbf{c}_k$ are produced by embedding class meta-data $\mathbf{v}_k$. In either case, embedded query points are classified via a softmax over distances to class prototypes: $p_\phi(y = k|\mathbf{x}) \propto \exp(-d(f_\phi(\mathbf{x}), \mathbf{c}_k))$.

Metric based learning

Learning to Compare : Relation Network for Few-shot Learning

- 기존 거리 계산 방식이 'fixed linear comparator' 였다면, 이 번에는 거리 계산에 뉴럴넷을 적용해 learnable non-linear comparator를 적용.
- 직관적이고 알기 쉬움



Learning to Compare: Relation Network for Few-Shot Learning

Flood Sung Yongxin Yang[3] Li Zhang[2] Tao Xiang[1] Philip H.S. Torr[2] Timothy M. Hospedales[3]
[1]Queen Mary University of London [2]University of Oxford [3]The University of Edinburgh
floodsung@gmail.com t.xiang@qmul.ac.uk
{lz, phst}@robots.ox.ac.uk {yongxin.yang, t.hospedales}@ed.ac.uk

Abstract

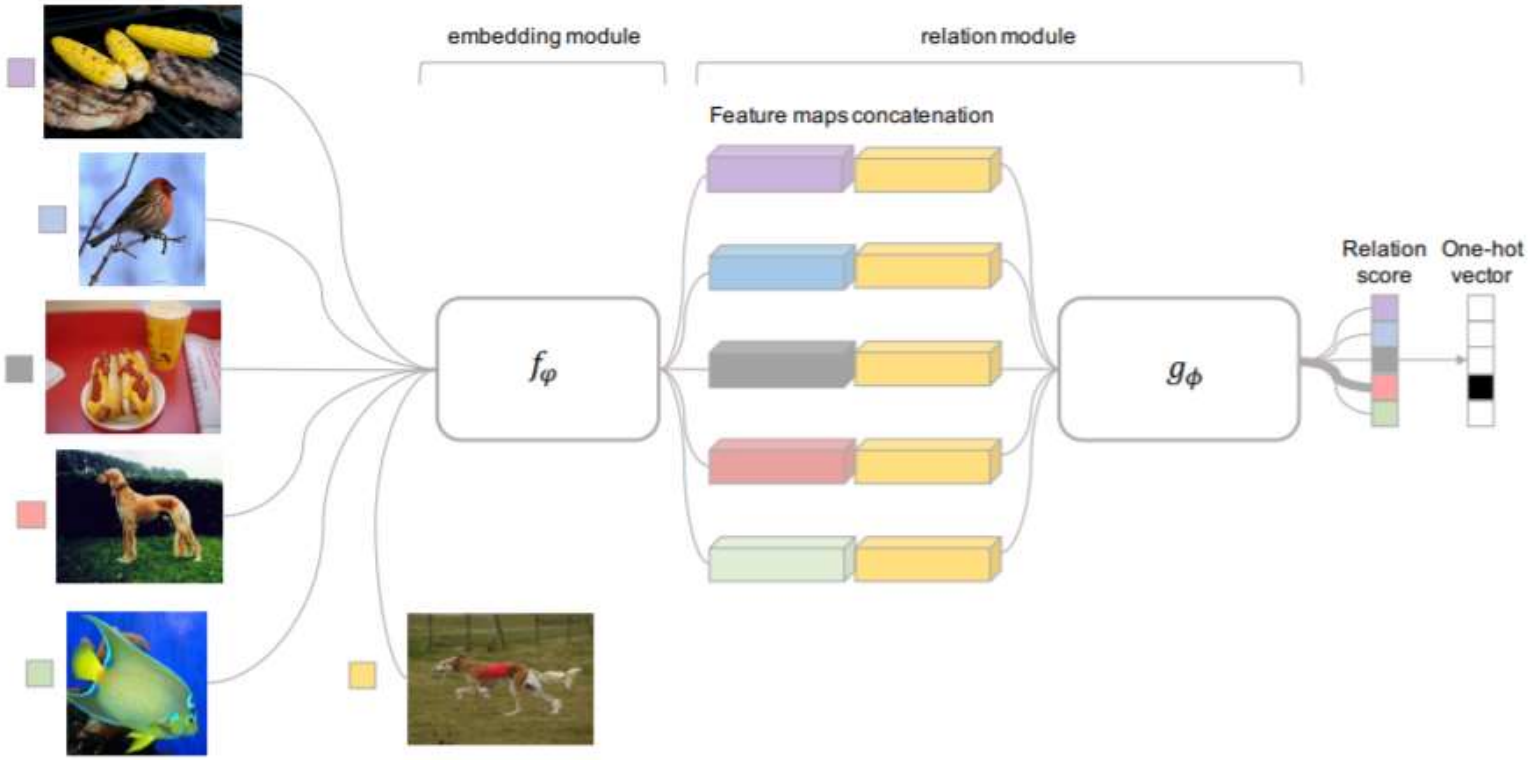
We present a conceptually simple, flexible, and general framework for few-shot learning, where a classifier must learn to recognise new classes given only few examples from each. Our method, called the Relation Network (RN), is trained end-to-end from scratch. During meta-learning, it learns to learn a deep distance metric to compare a small number of images within episodes, each of which is designed to simulate the few-shot setting. Once trained, a RN is able to classify images of new classes by computing relation scores between query images and the few examples of each new class without further updating the network. Besides providing improved performance on few-shot learning, our framework is easily extended to zero-shot learning. Extensive experiments on five benchmarks demonstrate that our simple approach provides a unified and effective approach for both of these two tasks.

zero-shot [11, 3, 24, 45, 25, 31] learning.

Few-shot learning aims to recognise novel visual categories from very few labelled examples. The availability of only one or very few examples challenges the standard 'fine-tuning' practice in deep learning [10]. Data augmentation and regularisation techniques can alleviate overfitting in such a limited-data regime, but they do not solve it. Therefore contemporary approaches to few-shot learning often decompose training into an auxiliary meta learning phase where transferrable knowledge is learned in the form of good initial conditions [10], embeddings [36, 39] or optimisation strategies [29]. The target few-shot learning problem is then learned by fine-tuning [10] with the learned optimisation strategy [29] or computed in a feed-forward pass [36, 39, 4, 32] without updating network weights. Zero-shot learning also suffers from a related challenge. Recognisers are trained by a single example in the form of a class description (c.f., single exemplar image in one-shot), making data insufficiency for gradient-based learning a challenge.

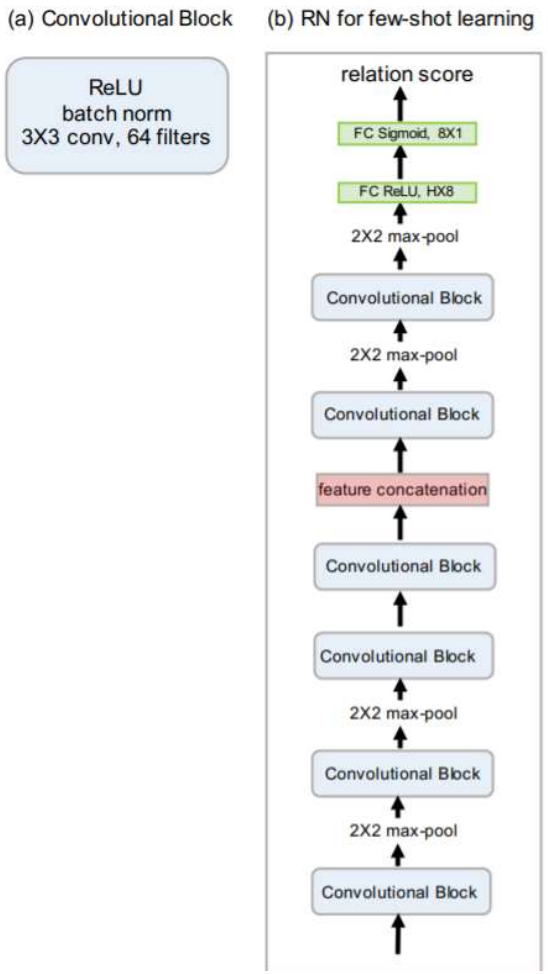
Metric based learning

Learning to Compare : Relation Network for Few-shot Learning



Metric based learning

Learning to Compare : Relation Network for Few-shot Learning



Optimization based Learning

Next…

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

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

Abstract

We propose an algorithm for meta-learning that is model-agnostic, in the sense that it is compatible with any model trained with gradient descent and applicable to a variety of different learning problems, including classification, regression, and reinforcement learning. The goal of meta-learning is to train a model on a variety of learning tasks, such that it can solve new learning tasks using only a small number of training samples. In our approach, the parameters of the model are explicitly trained such that a small number of gradient steps with a small amount of training data from a new task will produce good generalization performance on that task. In effect, our method trains the model to be easy to fine-tune. We demonstrate that this approach leads to state-of-the-art performance on two few-shot image classification benchmarks, produces good results on few-shot regression, and accelerates fine-tuning for policy gradient reinforcement learning with neural network policies.

the form of computation required to complete the task.

In this work, we propose a meta-learning algorithm that is general and model-agnostic, in the sense that it can be directly applied to any learning problem and model that is trained with a gradient descent procedure. Our focus is on deep neural network models, but we illustrate how our approach can easily handle different architectures and different problem settings, including classification, regression, and policy gradient reinforcement learning, with minimal modification. In meta-learning, the goal of the trained model is to quickly learn a new task from a small amount of new data, and the model is trained by the meta-learner to be able to learn on a large number of different tasks. The key idea underlying our method is to train the model's initial parameters such that the model has maximal performance on a new task after the parameters have been updated through one or more gradient steps computed with a small amount of data from that new task. Unlike prior meta-learning methods that learn an update function or learning rule (Schmidhuber, 1987; Bengio et al., 1992; Andrychowicz et al., 2016; Ravi & Larochelle, 2017), our algorithm does not expand the number of learned param-

Learning to Forget for Meta-Learning

Sungyong Baik Seokil Hong Kyoung Mu Lee
ASRI, Department of ECE, Seoul National University
{dsybaik, hongceo96, kyoungmu}@snu.ac.kr

Abstract

Few-shot learning is a challenging problem where the goal is to achieve generalization from only few examples. Model-agnostic meta-learning (MAML) tackles the problem by formulating prior knowledge as a common initialization across tasks, which is then used to quickly adapt to unseen tasks. However, forcibly sharing an initialization can lead to conflicts among tasks and the compromised (undesired by tasks) location on optimization landscape, thereby hindering the task adaptation. Further, we observe that the degree of conflict differs among not only tasks but also layers of a neural network. Thus, we propose task-and-layer-wise attenuation on the compromised initialization to reduce its influence. As the attenuation dynamically controls (or selectively forgets) the influence of prior knowledge for a given task and each layer, we name our method as L2F (Learn to Forget). The experimental results demonstrate that the proposed method provides faster adaptation and greatly improves the performance. Furthermore, L2F can be easily applied and improve other state-of-the-art MAML-based frameworks, illustrating its simplicity and generalizability.

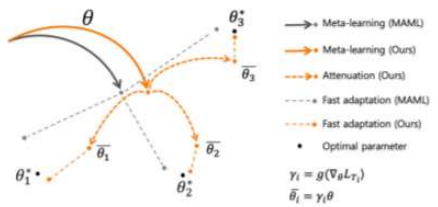


Figure 1: When there is a large degree of *conflict*, the updated initialization ends up in the location neither of tasks desires. Such undesired (hence compromised) initialization location can make learning difficult during fast adaptation to each task. Our method makes the fast adaptation easier by minimizing the influence of the compromised initialization for each task, through attenuation parameter γ generated by the task-conditioned network g. This makes the optimization landscape smoother and hence helps achieve better generalization to unseen examples.