CDFSL

Cross-Domain Few-Shot Learning

AMMAI FINAL PROJECT

r08944004 萬世澤
r08725047 古佳峻

Outline

Motivation

Few-Shot Task

Cross-Domain Few-Shot Task

Conclusion

Motivation

Our consideration

- In addition to leveraging the metric learning methods, we would like to delve into the contents more about manifold learning.
- Especially, learn a real case about how to utilize the manifold structure of data with deep learning techniques actually.

Combine idea

- For few-shot task, we combine the data augmentation and the manifold learning.
- For cross-domain few-shot task, we combine the manifold learning and the domain adaptation tricks.

Few-Shot Task

Our main approach utilizes the transductive propagation network, **TPN**[1].

Rely on Label Propagation Algorithm[2]

- Propagate neighbor's label by utilizing the connections in the labeled set (i.e. support set)

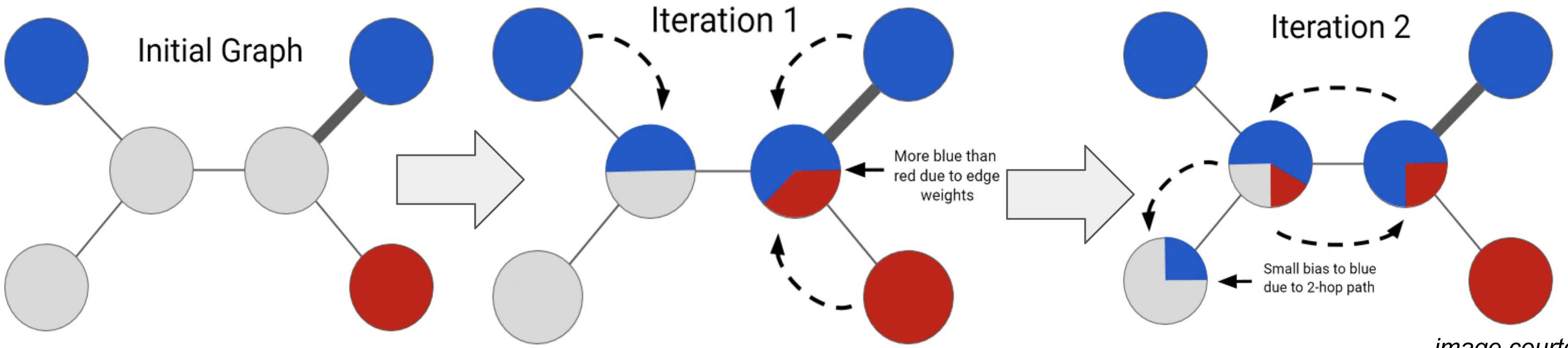
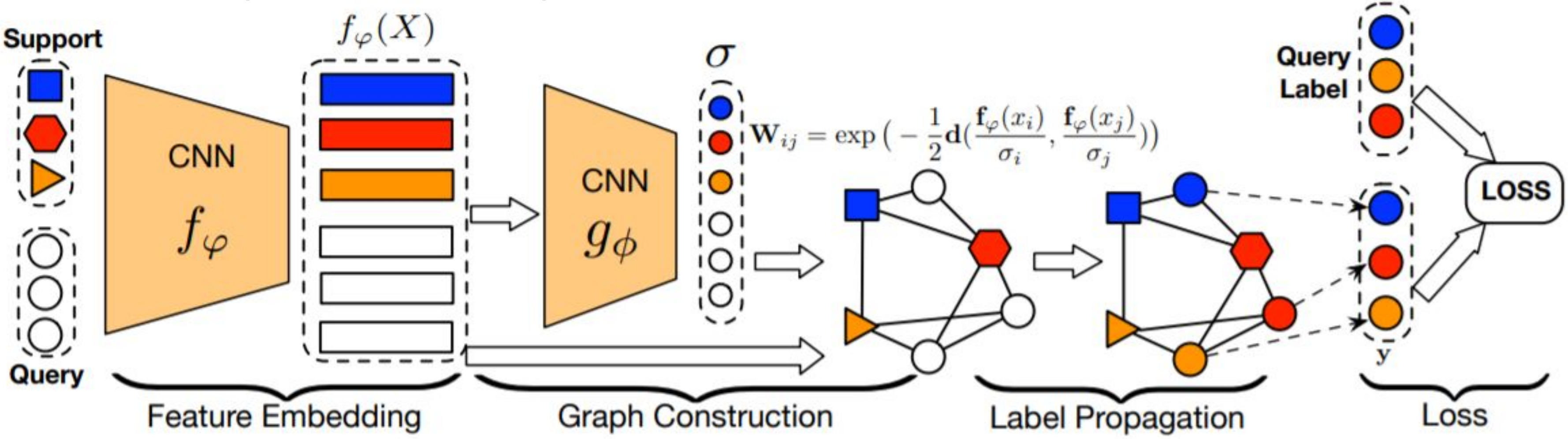


image courtesy of [CrowAI]

Propagate iteratively to determine the unknown labeled set (i.e. query set)

[1] Learning to Propagate Labels, ICLR 2019
[2] Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions, ICML 2003

Learning to propagate labels in TPN



Graph Construction

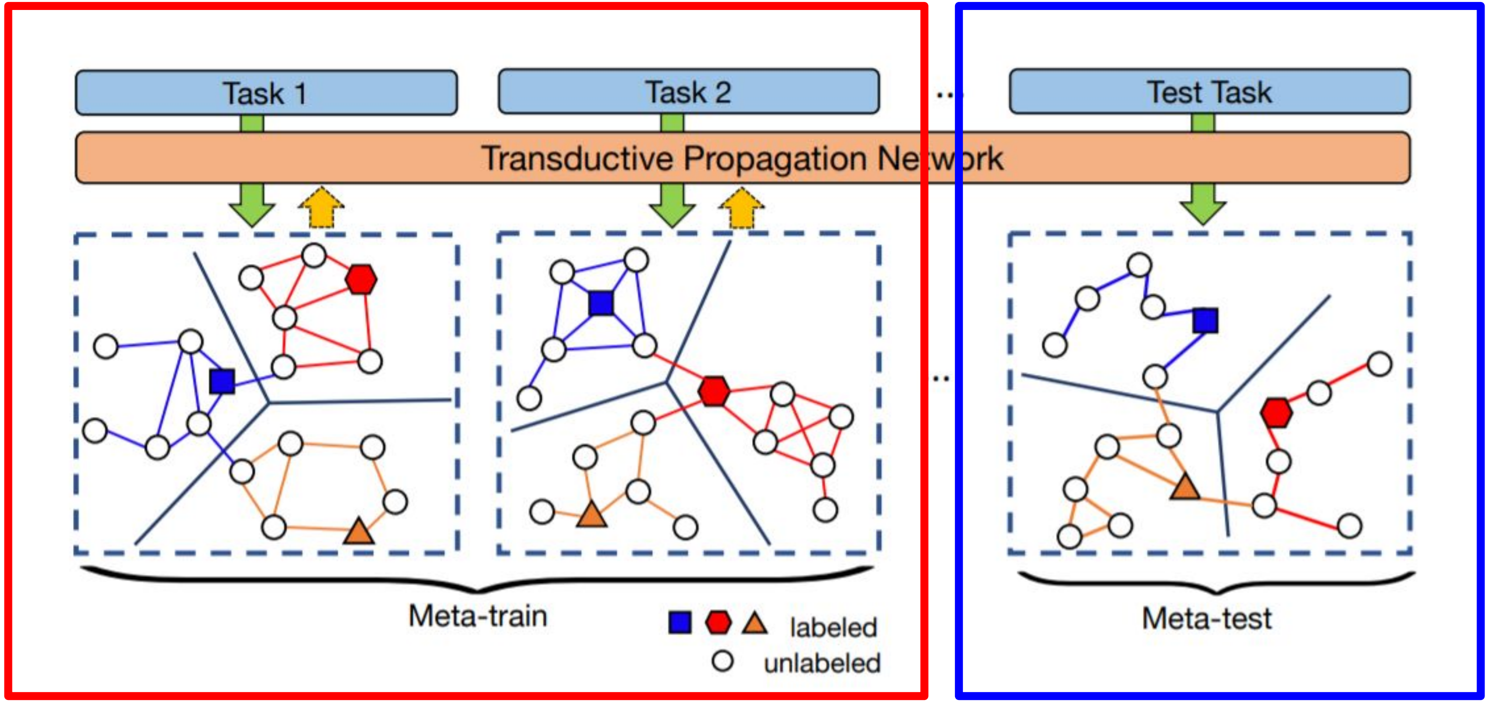
- Measure similarity for edge W using ”learnable” Gaussian similarity function.
- The parameter σ is learned in episodic training.

Note. Gaussian Similarity

$$W_{ij} = \exp\left(-\frac{d(\mathbf{x}_i, \mathbf{x}_j)}{2\sigma^2}\right)$$

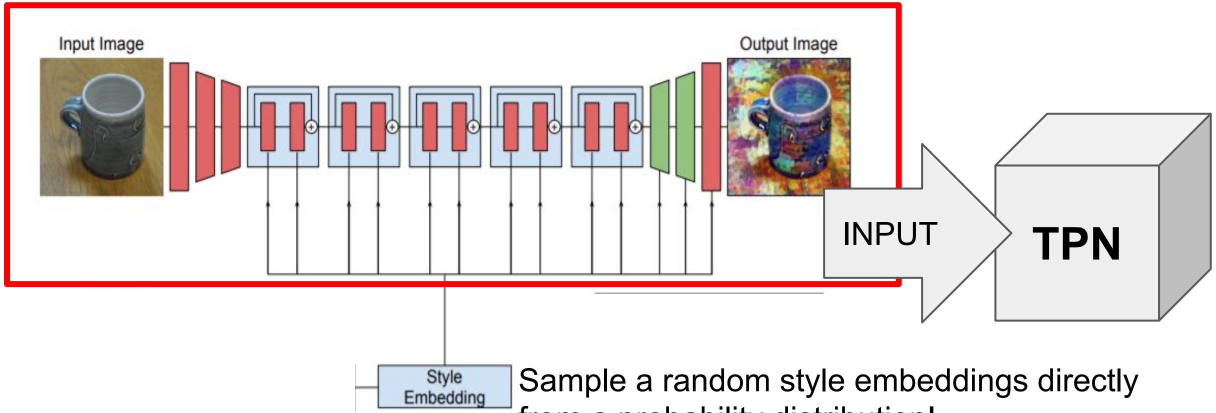
- **σ influences the behavior of label propagation.**
- No good way to tune σ in the past!

Few-Shot Learning using manifold structure with transductive inference



Utilize the **union of support set (labeled) and query set (unlabeled)** to exploit the manifold structure of novel class space.

Our Approach: learning to propagate labels using stylized data



Sample a random style embeddings directly from a probability distribution!

- Randomize texture, contrast and color, while preserving shape and semantic content.
- Style augmentation is only used during meta-training.

Style Augmentation: Data Augmentation via Style Randomization, CVPRW 2019

Experiments

	miniImageNet
Baseline (TA's)	**68.10 ± 0.67**
ProtoNet (TA's)	66.33 ± 0.65
Baseline (Ours)	**66.53 ± 0.66**
ProtoNet (Ours)	66.20 ± 0.67
ProtoNet + Angular Softmax[1] (margin=3)	69.32 +- 0.66
TPN (Liu et al., 2019)	**69.43**
TPN (Ours)	67.30 ± 0.65
Baseline + Style Augment	68.98 ± 0.65
ProtoNet + Style Augment	69.32 ± 0.68
ProtoNet + Angular Softmax + Style Augment	**69.64 ± 0.64**
TPN + Style Augment	65.44 ± 0.69

[1] SphereFace, CVPR 2017

unreproducible

metric learning
large-margin methods

manifold learning
(unreproducible)

May be not robust against noisy data points (Stylized data)

For a fair comparison, we use the same backbone (*ResNet10*), number of epochs(*400*), preprocessing tricks (*random crop, image jitter, and random horizontal flip*) in this task.

Cross-Domain Few-Shot Task

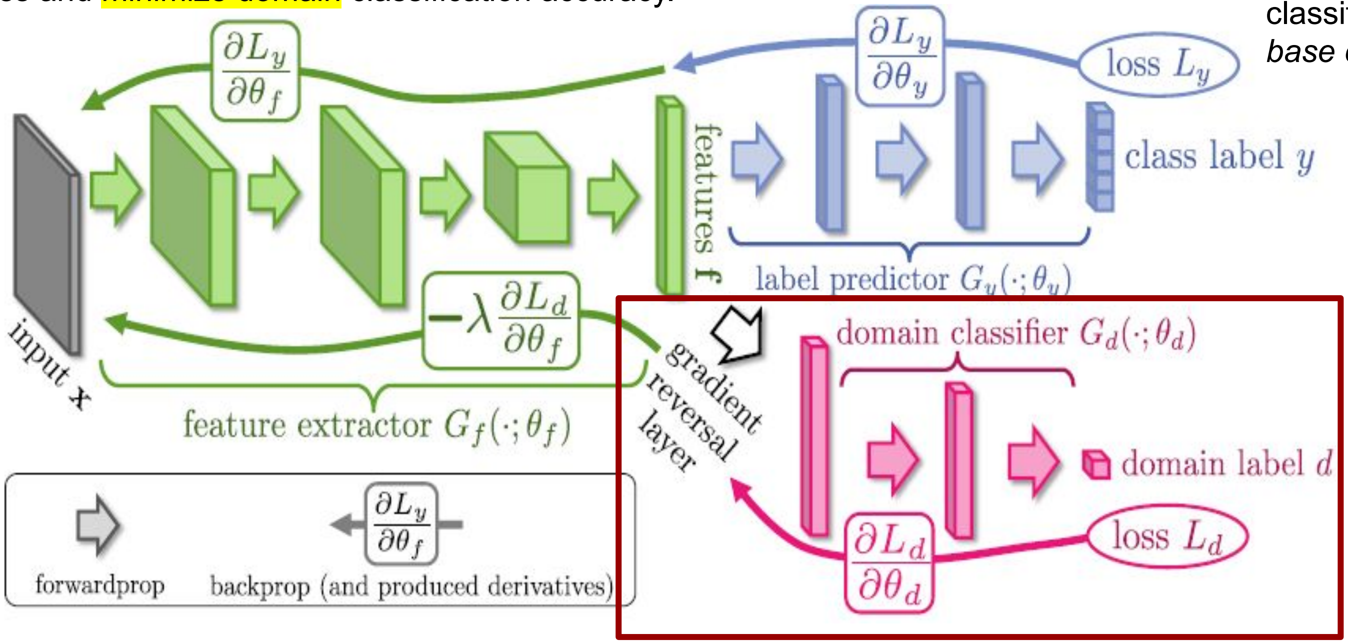
Unsupervised Cross-Domain Few-Shot Learning

- Assumption: Given an input x and the corresponding label y in the novel class, we assume the novel class input x can be learned with the base class input x' jointly in meta-training stage. (Note. novel class label y won't be used!)
- Our goal is to make the feature extractor learn domain-invariant features.
  - Inspired by domain-adversarial training, we apply **gradient reversal** techniques.

Gradient Reversal

Green one maximize label classification accuracy in the *base class* and minimize domain classification accuracy.

Blue one maximize label classification accuracy in the *base class*

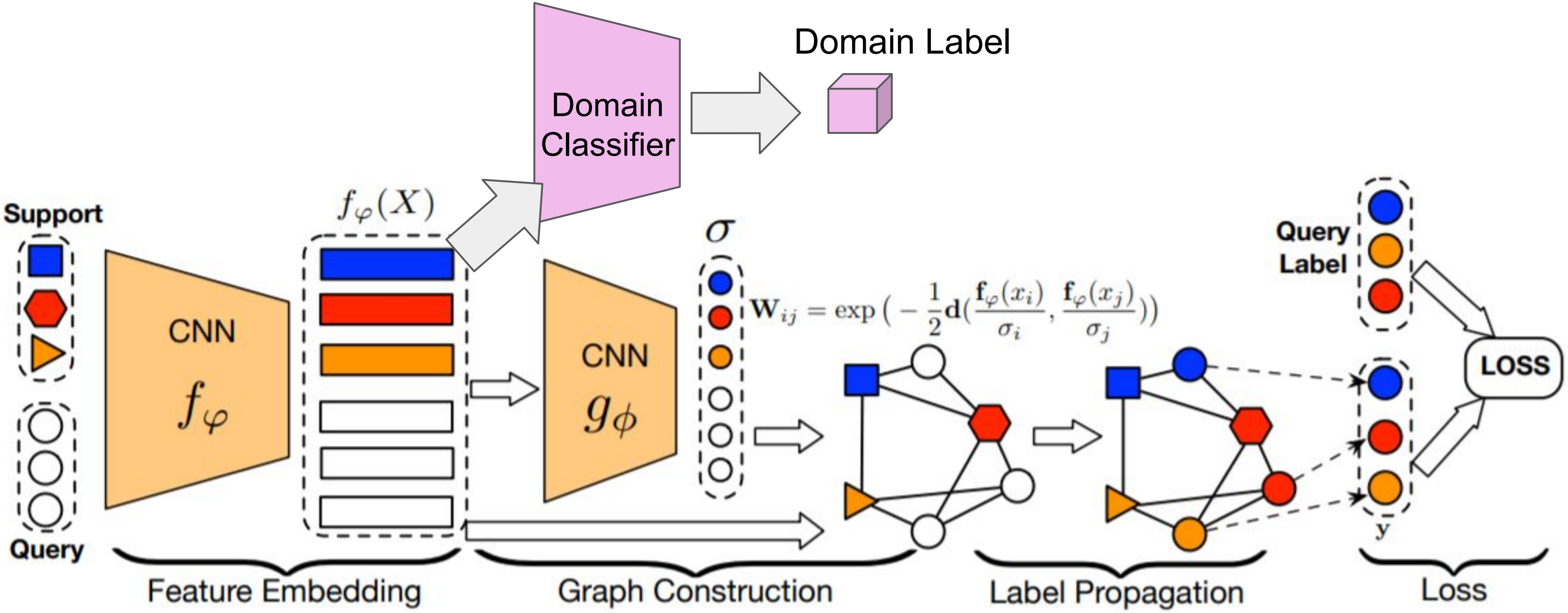


Red one maximize domain classification accuracy.

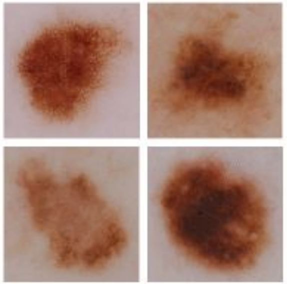
Goal: Differentiate the instances from *novel class* or *base class*!

Unsupervised Domain Adaptation by Backpropagation, ICML 2015

Our Approach: learn to propagate labels using domain-invariant features



Experiments



	EuroSAT	ISIC
Baseline	76.37 ± 0.69 / 78.95 ± 0.62	44.31 ± 0.58 / **48.87 ± 0.62**
ProtoNet	78.44 ± 0.67 / 82.09 ± 0.63	42.12 ± 0.56 / 46.93 ± 0.58
TPN	64.41 ± 0.80 / 82.07 ± 0.67	33.86 ± 0.54 / 44.09 ± 0.60
TPN (miniImageNet->ISIC)	63.23 ± 0.73 / 82.33 ± 0.67	31.93 ± 0.55 / 44.01 ± 0.57
TPN (miniImageNet->EuroSAT)	58.04 ± 0.67 / **82.61 ± 0.62**	32.79 ± 0.50 / 44.38 ± 0.59

For a fair comparison, we use the same backbone(*ResNet10*), number of epochs(*400*), preprocessing tricks(*random crop, image jitter, and random horizontal flip*) in this task.
The result w/o and w/ fine-tuning are splitted by slash.

Unsupervised domain-adaptation meta-training using *miniImageNet* and *ISIC*!

Conclusion

1. Learn about the real case combining manifold learning and deep learning.
2. Using style augmentation, large-margin methods and manifold structure is helpful for improving accuracy for few-shot task.
3. Combining manifold learning and gradient reversal tricks may not be helpful for cross-domain settings.