TAdaNet: Task-Adaptive Network for Graph-Enriched Meta-Learning

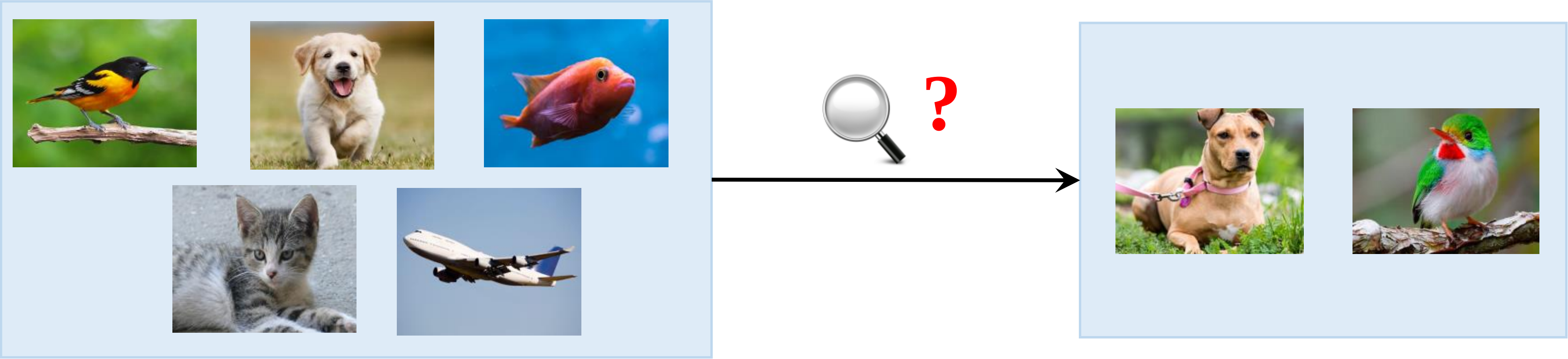
Qiuling Suo[1], Jingyuan Chou[2], Weida Zhong[1], Aidong Zhang[2]

[1]University at Buffalo, [2]University of Virginia

Introduction

Limited Training Data

- Image Classification



Training images

Testing images

- Disease Prediction

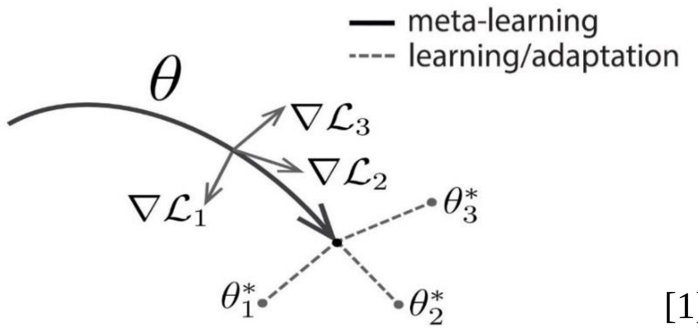
Patients with different diseases

New patients

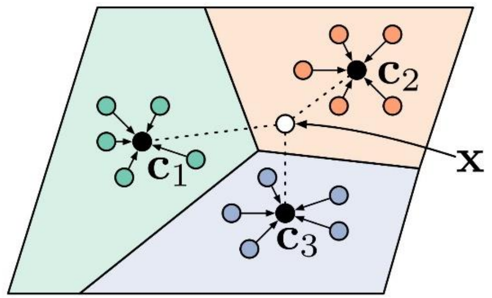
Introduction

- **Meta-learning**
  - Learn transferable knowledge from multiple training tasks and generalize to new tasks with limited supervised experience
  - An effective approach for few-shot learning

- **Popular methods**
  - Learn global initializations, metric or optimizers
  - Assume the globally shared information can be transferred across all tasks



[1]



[2]

[1] Finn Chelsea, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. ICML 2017.
[2] Snell J, Swersky K, Zemel R. Prototypical networks for few-shot learning. NeurIPS 2017.

Challenges

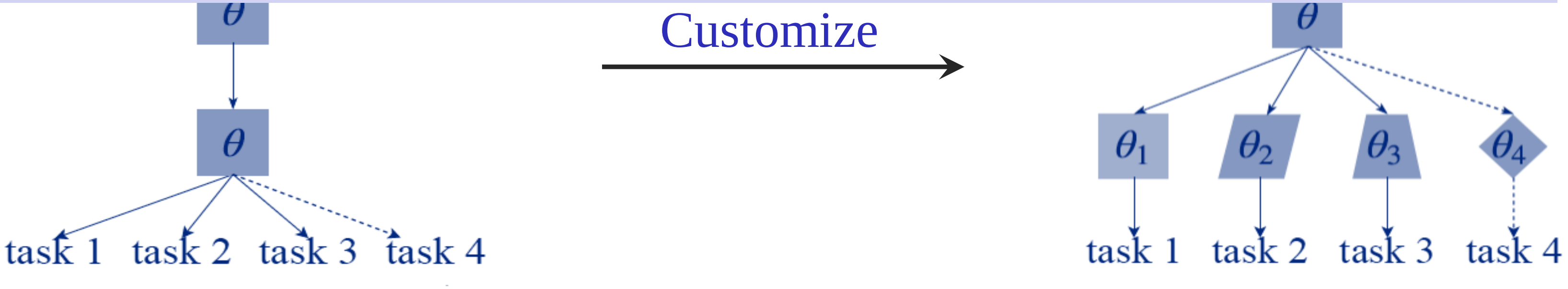
■ Task heterogeneity

- Task distributions can be diverse
- Global parameters may not well handle tasks with different underlying distributions

■ Recent approaches

- Learn task embeddings by aggregating data examples or handcrafted structure
- Customize global initializations or metric with task-specific conditioning
- Rely purely on data itself to learn task-relationships

How to effectively capture and utilize task relationships?



Huaxiu Yao, Ying Wei, Junzhou Huang, Zhenhui Li. Hierarchically Structured Meta-learning. ICML 2019

Vuorio R, Sun SH, Hu H, Lim JJ. Multimodal Model-Agnostic Meta-Learning via Task-Aware Modulation. NeurIPS 2019.

External Knowledge

- Domain knowledge often exhibits in the form of graphs.
- In the meta-learning setting, a task contains several classes that are represented as nodes in the graph.

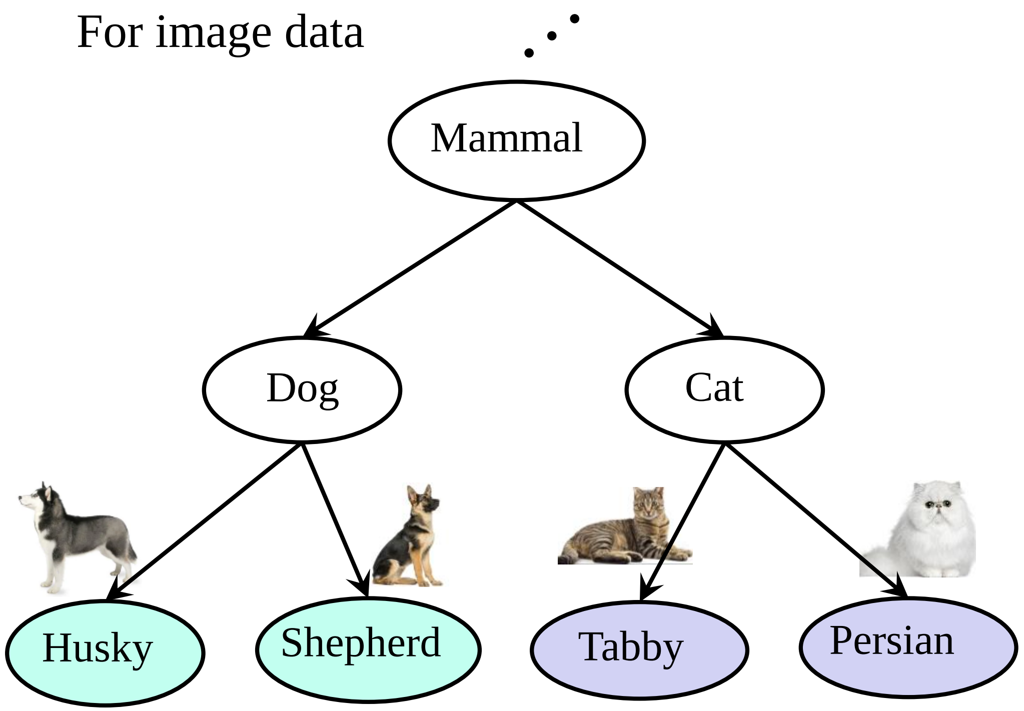
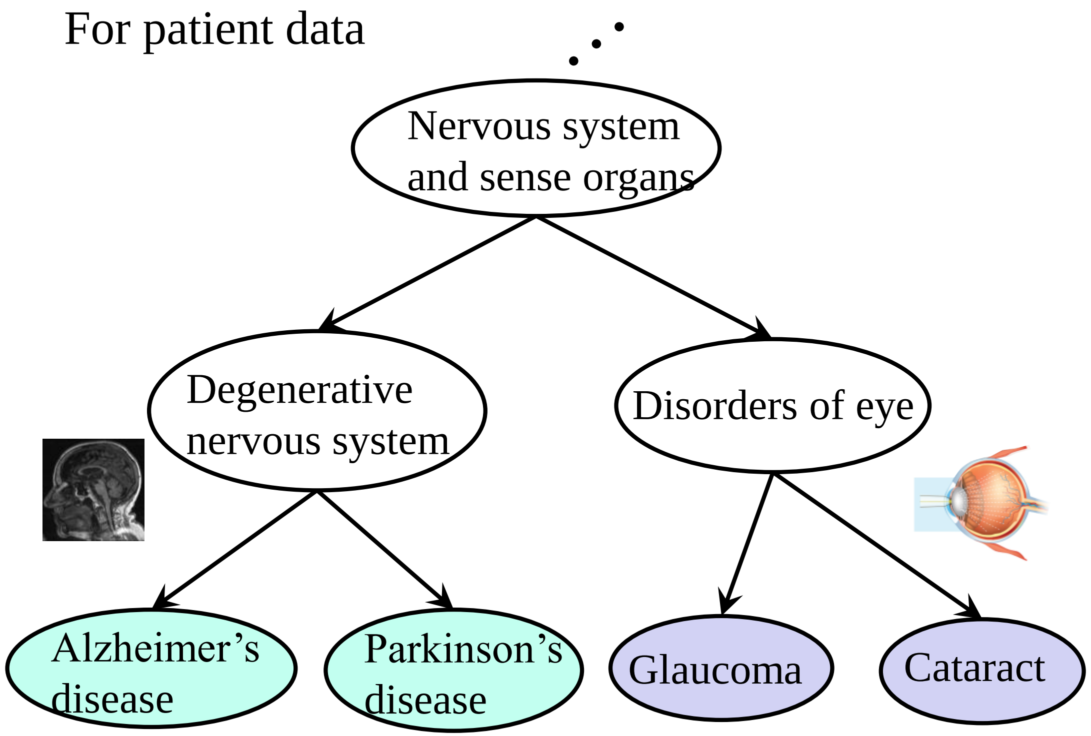


Image category hierarchy



Disease ontology

Our Method

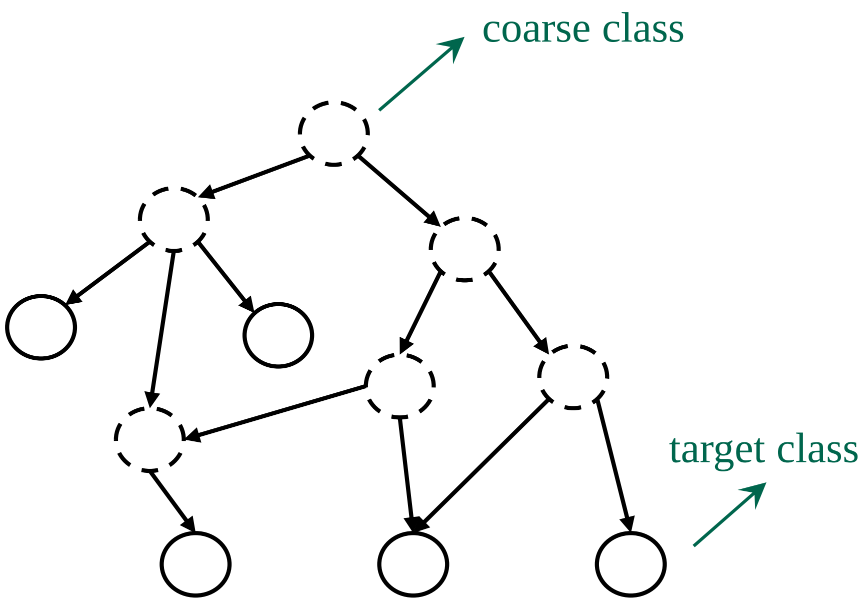
- Incorporate domain knowledge represented in the graph

- Enrich data representation
  - Allow message passing across nodes of the given category graph for representation learning

- Enhance task relationships
  - Produce task-aware parameter adjustment based on task embeddings

Method: Problem Setting

- A task $\mathcal{T}_i$
  - Sampled from a complex distribution $p(\mathcal{T})$
  - Contain support set D_i^{tr} and query set D_i^{te}
  - N-way K-shot classification

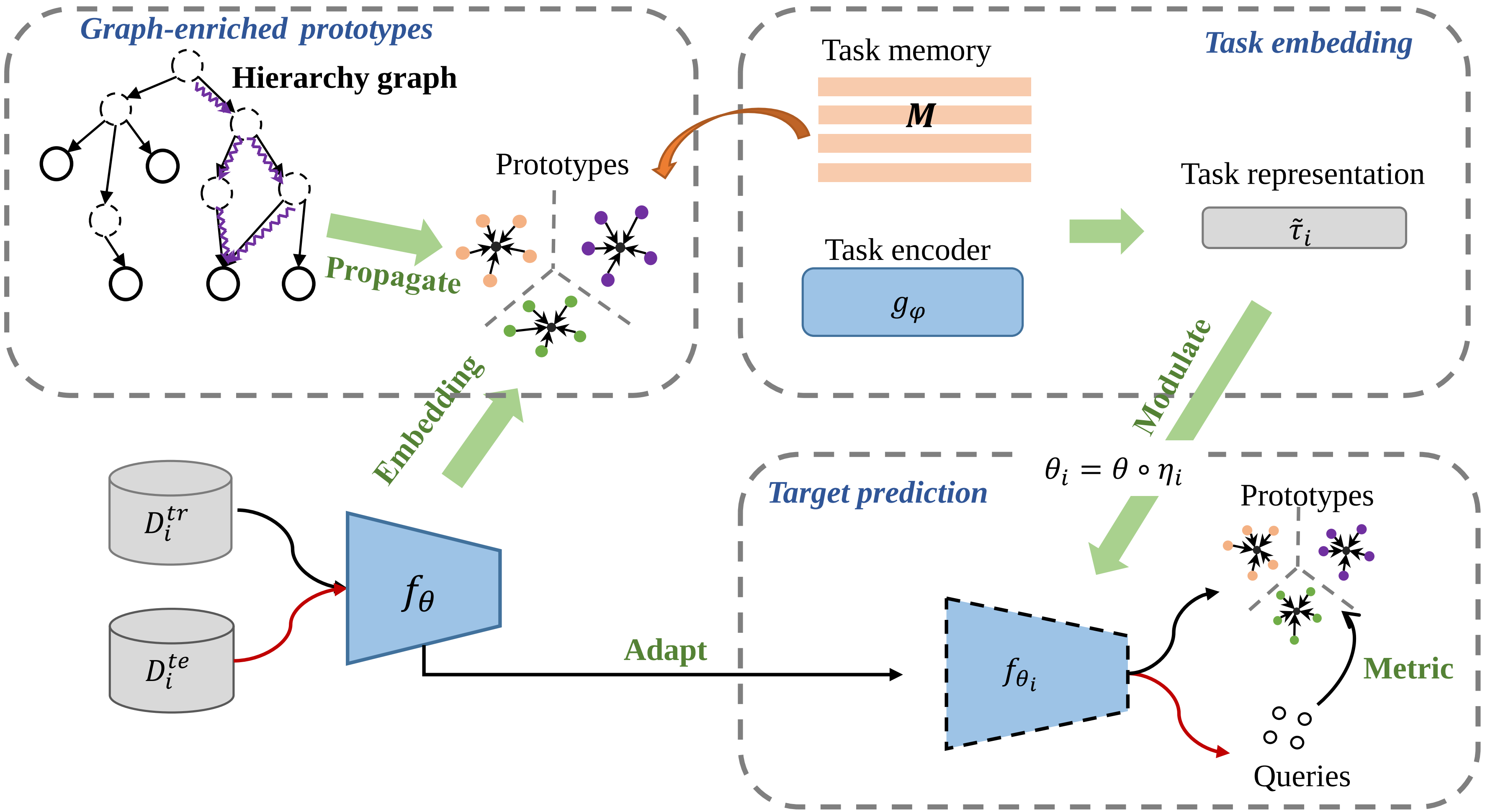


- Hierarchy graph
  - Node: class
  - Edge: parent -> child relationship
  - Leaf nodes: few-shot target classes
  - Ancestor nodes: coarse classes

- Assumptions
  - A task is considered more similar to another task sharing nodes in the graph, than the one that has disjoint classes with it.
  - Similar tasks should share more information and have similar model parameters.

Method: TAdaNet

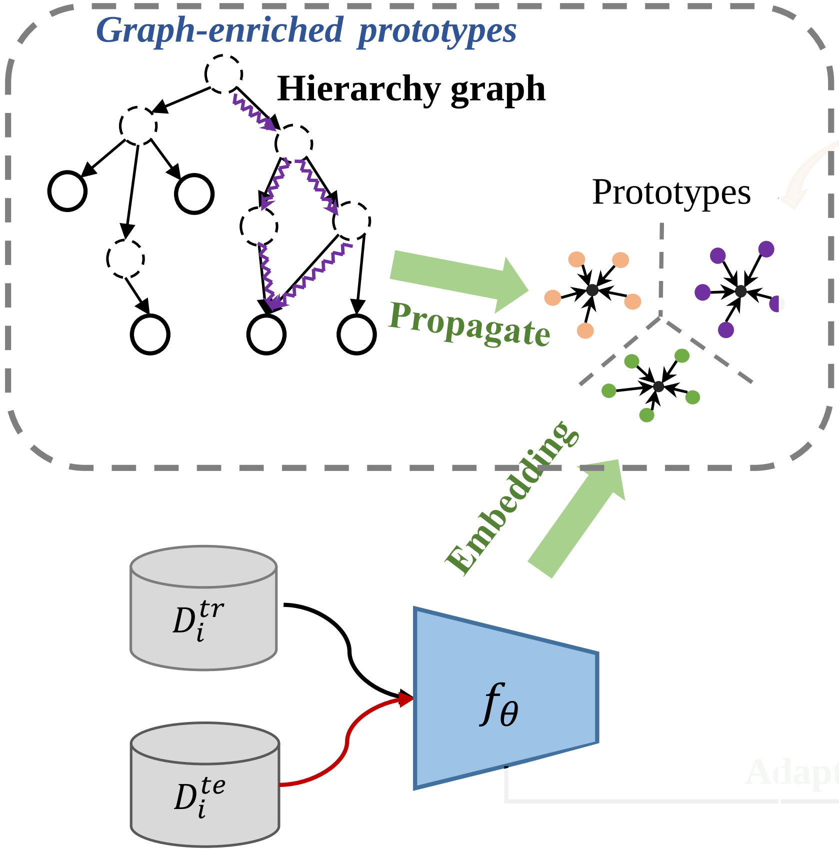
Overall framework



- Leverage the information on hierarchy graph and transferable knowledge from related tasks
- Learn task-adaptive metric space for target prediction

Method: TAdaNet

- Learn graph-enriched prototype representations



- Initial prototype $c_i^{k,0}$ representation for a class k
- Aggregate neighbor embeddings through graph attention to obtain graph-enriched prototype $c_i^{k,L}$
- Obtain updated prototype c_i^k

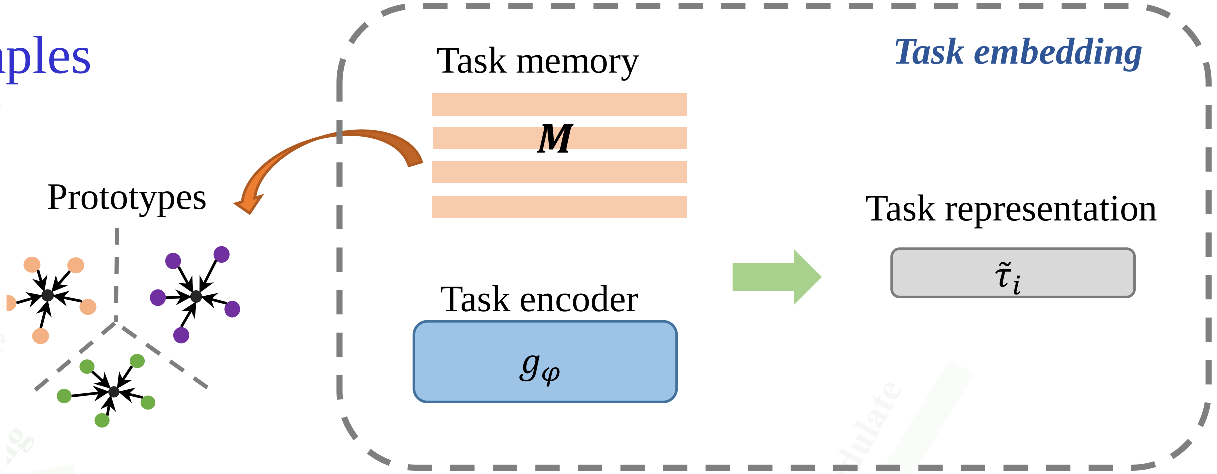
$$c_i^k = (1 - \lambda)c_i^{k,0} + \lambda c_i^{k,L}$$

Method: TAdaNet

Learn task-embeddings

- Aggregate support examples in each task with mean pooling

$$\tau_i = \frac{1}{NK}\sum g_\phi(x_{i,j})$$



- Regularize the embeddings not far away from graph-enriched prototypes

$$\mathcal{L}_c(\mathcal{T}_i) = \sum |c_i^k - \frac{1}{n_i^{tr,k}}\sum g_\phi(x_{i,j})|_F^2$$

- Retrieve information from memory net M, and obtain memory enhanced task embedding $\tilde{\tau}_i$

Target prediction

- Task-adaptive parameter gate

$$\eta_i = F_c(\tilde{\tau}_i)$$

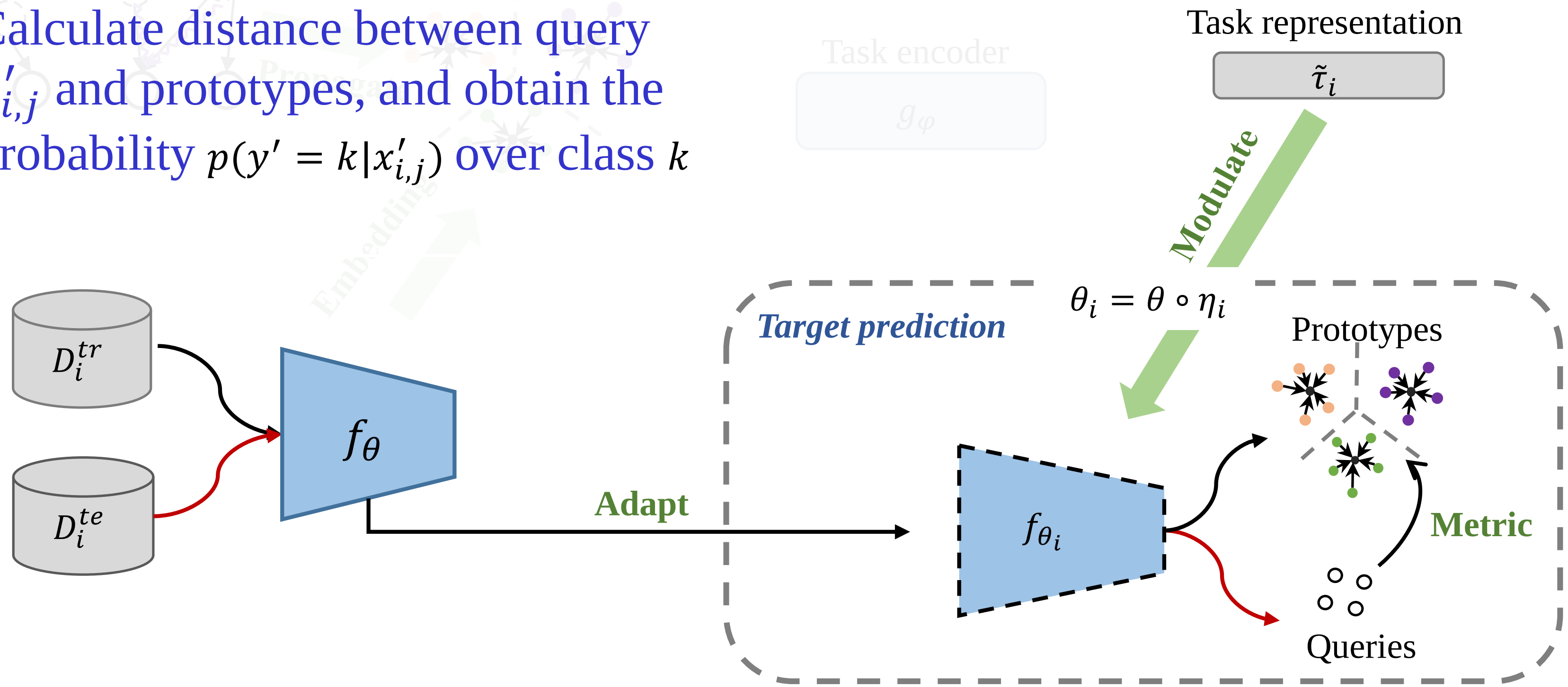
- Customized parameter

$$\theta_i = \theta \circ \eta_i$$

- Calculate distance between query $x'_{i,j}$ and prototypes, and obtain the probability $p(y' = k|x'_{i,j})$ over class k

- Classification loss

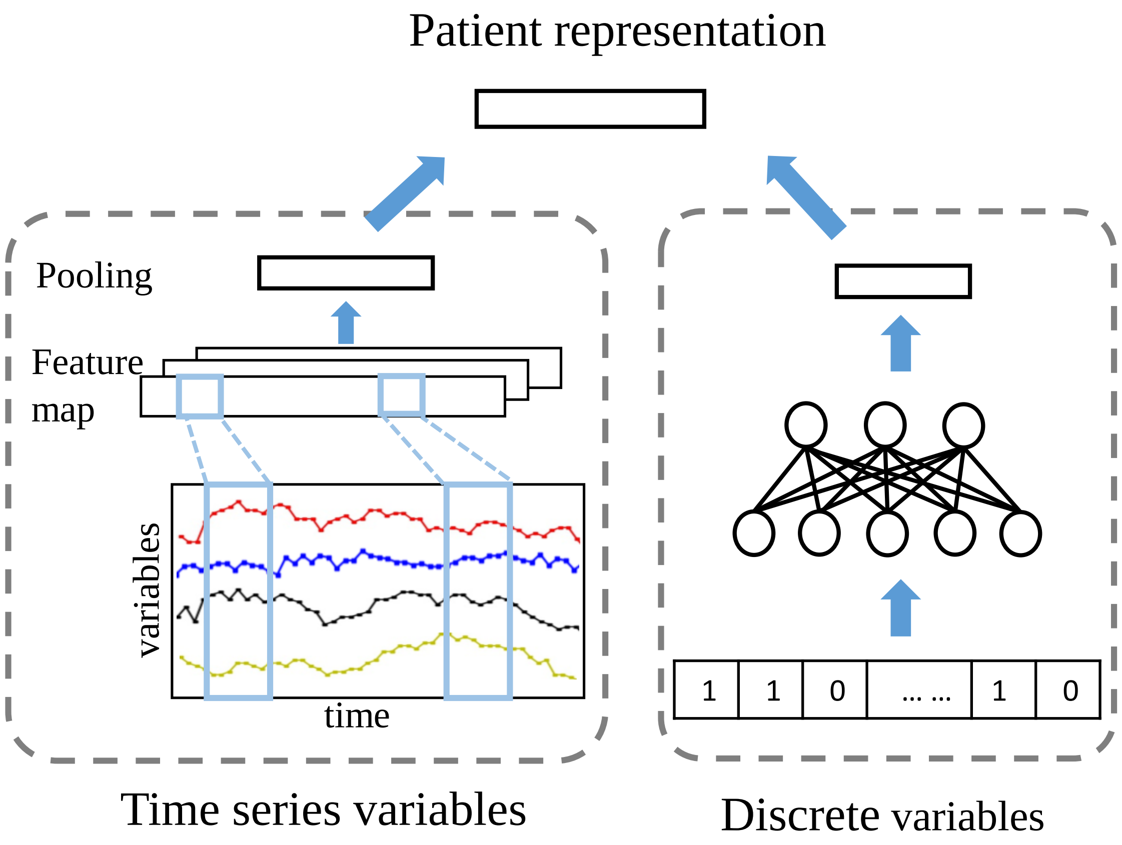
$$\mathcal{L}_p(\mathcal{T}_i) = -\sum \log p(y' = k|x'_{i,j})$$



Method: Backbone Networks

- **Image classification**
  - 4 convolutional layers, each with 64 filters of kernel size 3

- **Disease detection**
  - Time series variables: one-directional convolution operation
  - Discrete variables (ICD-9 codes): fully-connected layers
  - Combine the two learned vectors to obtain the representation for each patient



Experiments

- **Datasets:**
  - processed from *tiered*ImageNet and MIMIC-III separately

Table: Statistics of two datasets

Dataset		training		testing	
		#classes	#examples	#classes	#examples
Image-graph	coarse	388	55,800	190	13,580
	target	441	8,820	113	2,260
	total	829	64,620	303	15,840
MIMIC	coarse	155	11,099	74	4,731
	target	171	11,428	46	2,902
	total	326	22,527	120	7,651

- **Baselines**
  - Metric-based: protoNet, matchingNet, relationNet, PPN
  - Gradient-based: MAML, MMAML, HSML

- **Task sampling strategies**
  - Subgraph sample: each task is sampled from one subgraph; heterogeneous tasks
  - Random sample: randomly select from leaf nodes to form a task

Experiment: results on image classification

Methods	Subgraph sampling			Random sampling		
	5-way 1-shot	5-way 5-shot	10-way 1-shot	5-way 1-shot	5-way 5-shot	10-way 1-shot
MAML	38.12±1.06%	53.82±1.14%	21.15±0.91%	47.50±1.04%	60.90±1.10%	32.70±0.67%
MMAML	39.60±0.84%	54.23±1.19%	23.05±1.16%	47.34±1.05%	61.96±0.88%	32.40±1.20%
HSML	38.14±1.01%	54.85±1.08%	21.36±1.03%	46.56±1.01%	62.05±0.98%	32.68±0.66%
MatchingNet	38.21±1.08%	50.64±1.07%	25.16±0.75%	47.37±1.04%	64.03±0.81%	33.87±0.64%
ProtoNet	38.50±1.03%	54.31±1.06%	25.29±0.63%	46.95±1.04%	65.73±0.96%	33.65±0.69%
RelationNet	36.46±1.06%	51.87±1.01%	23.40±0.73%	49.94±1.06%	65.89±0.88%	34.16±0.64%
PPN	45.08±1.04%	53.32±1.05%	36.32±1.23%	56.35±1.01%	65.90±0.92%	49.47±1.31%
TAdaNet	**48.85±1.17%**	**55.29±1.13%**	**38.55±1.31%**	**60.01±0.98%**	**69.38±0.90%**	**52.20±1.24%**

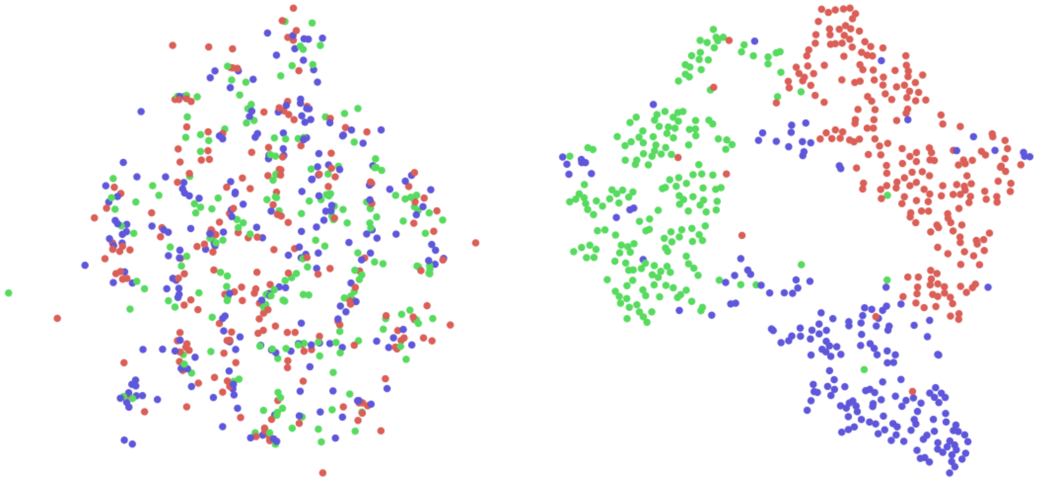
- Performing classification under subgraph sampling is more difficult than under random sampling.
- Graph information is especially helpful for 1-shot learning.

Experiment: results on disease classification

Methods	Subgraph sampling			Random sampling		
	3-way 1-shot	3-way 5-shot	5-way 1-shot	3-way 1-shot	3-way 5-shot	5-way 1-shot
MAML	40.54±0.95%	45.47±0.94%	26.29±0.61%	46.20±0.97%	53.70±0.94%	31.97±0.74%
MMAML	41.21±0.79%	46.32±1.01%	26.94±0.62%	45.91±0.98%	54.93±0.90%	32.25±0.78%
HSML	40.97±0.41%	45.76±0.86%	27.01±0.70%	45.94±0.84%	53.14±0.41%	31.04±0.70%
MatchingNet	39.12±0.70%	43.26±0.80%	26.08±0.57%	43.87±0.77%	50.92±0.92%	29.64±0.63%
ProtoNet	38.68±0.68%	46.24±0.97%	25.79±0.54%	42.92±0.72%	54.08±1.00%	29.61±0.65%
RelationNet	39.00±0.87%	42.46±0.96%	21.38±0.43%	43.85±0.89%	52.23±0.95%	28.67±0.61%
PPN	45.59±0.85%	50.55±0.99%	30.67±0.67%	51.54±0.90%	58.16±0.93%	38.59±0.68%
TAdaNet	**49.74±0.92%**	**52.05±0.91%**	**32.56±0.67%**	**54.06±0.94%**	**59.05±0.92%**	**40.31±0.72%**

- Performing classification under subgraph sampling is more difficult than under random sampling.
- Graph information is especially helpful for 1-shot learning.

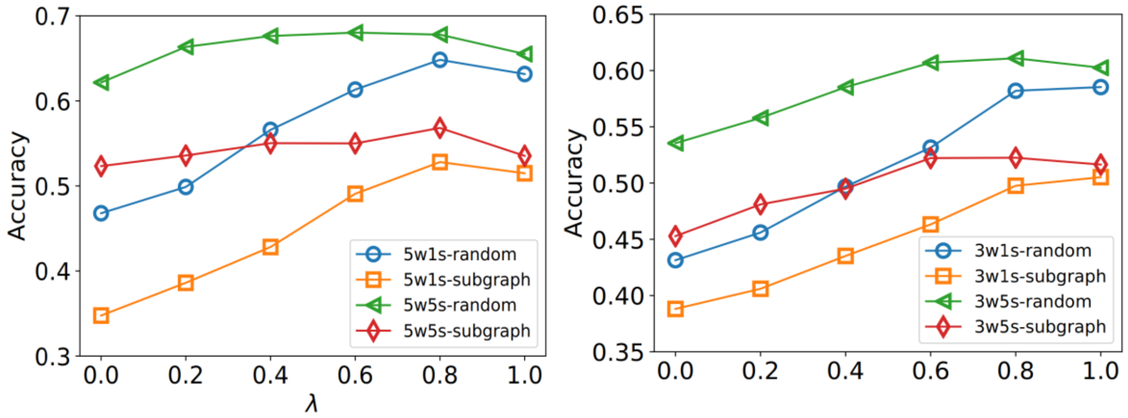
Experiment: analysis



(a) HSML (b) TAdaNet

Figure: Visualization of task embeddings.

- TAdaNet learns more accurate task embeddings than HSML.



(a) Image-graph (b) MIMIC

- λ is to balance prototypes learned from example mean and neighbors aggregation.

Huaxiu Yao, Ying Wei, Junzhou Huang, Zhenhui Li. Hierarchically Structured Meta-learning. ICML 2019

Thank You!