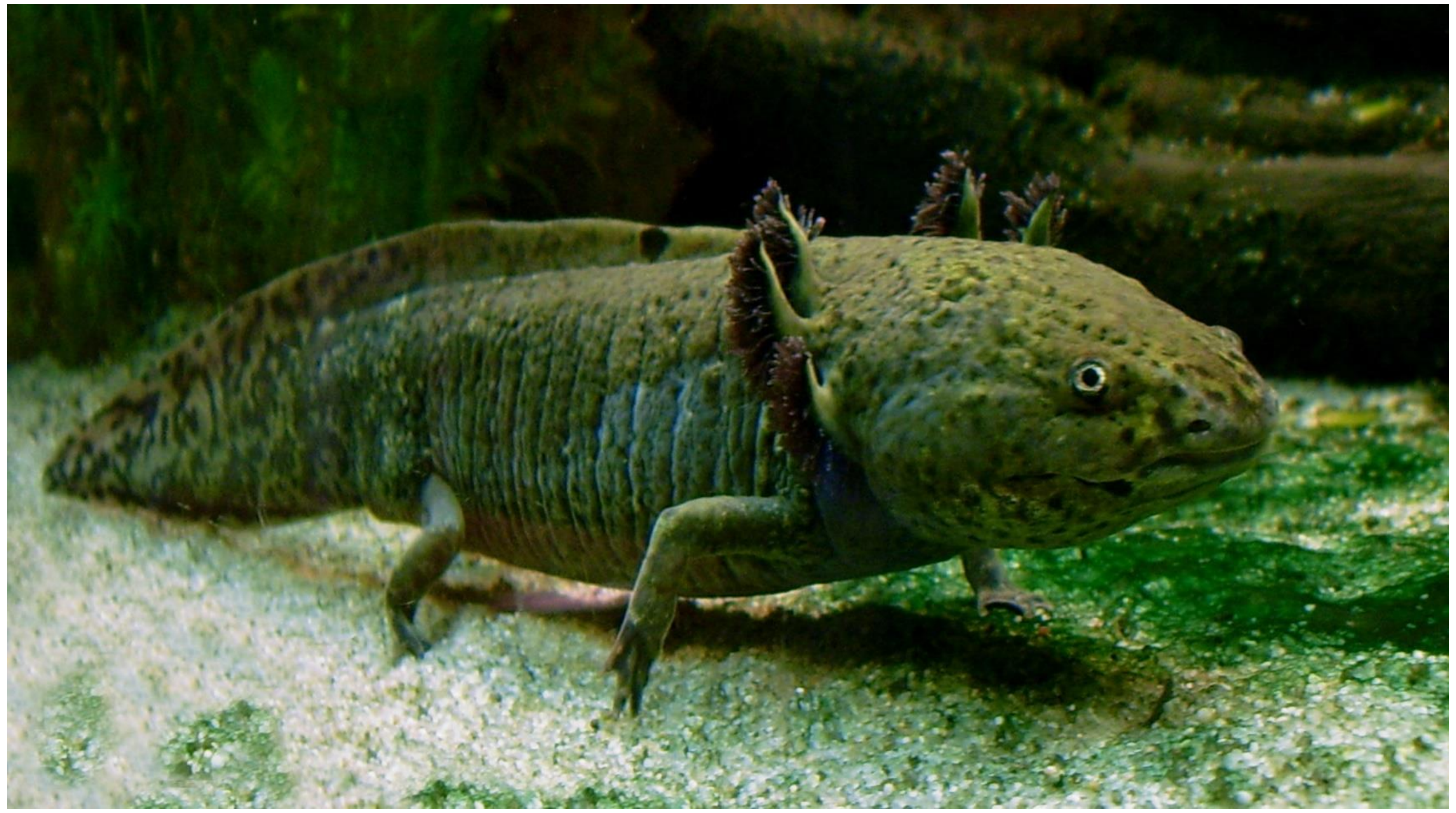
Research Project Proposal: Structured Meta-Learning for Heterogeneous Tasks

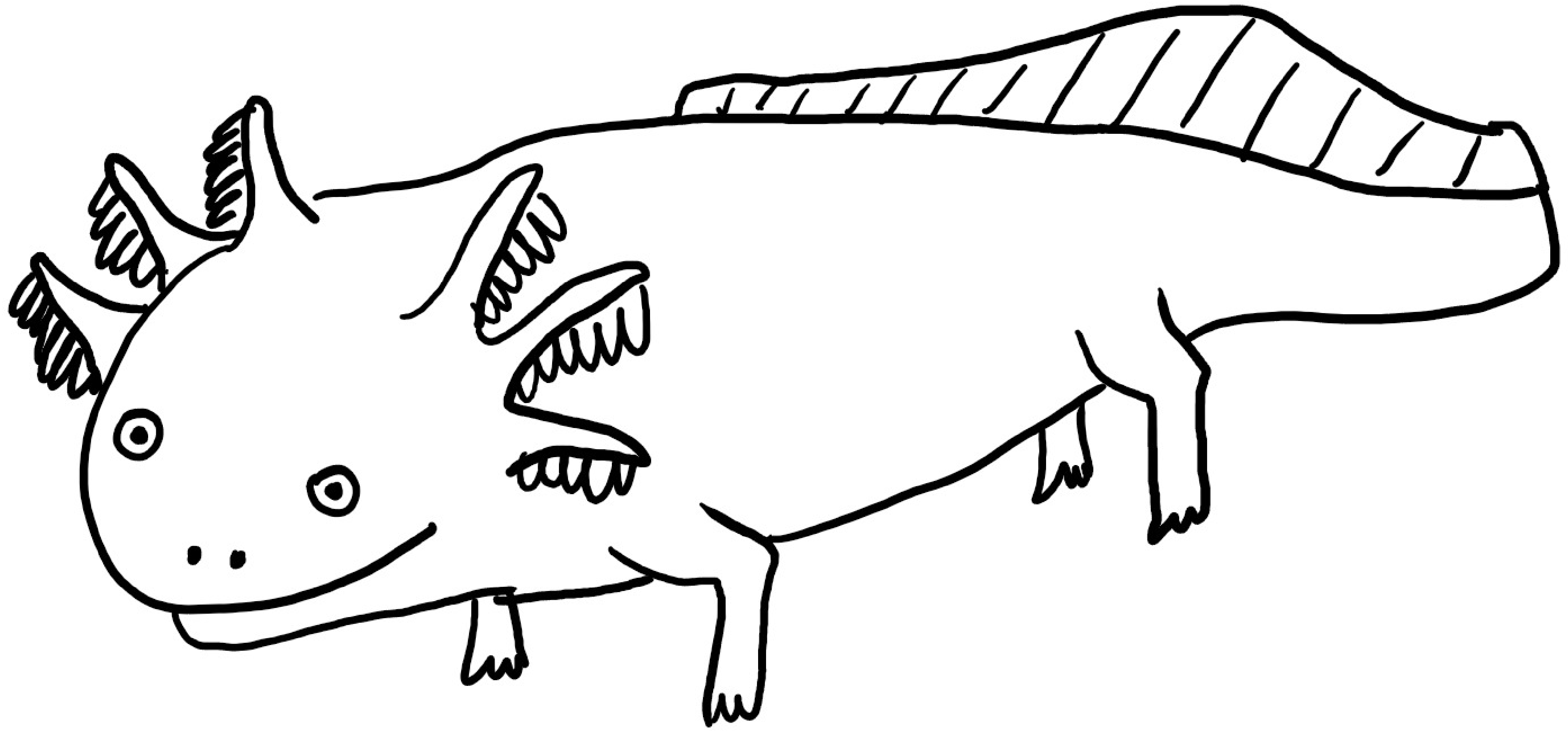
Nicola De Angeli
nicola.deangeli@mail.polimi.it
Computer Science and Engineering Track
Advisors: Matteo Matteucci, Marco Ciccone

POLITECNICO MILANO 1863

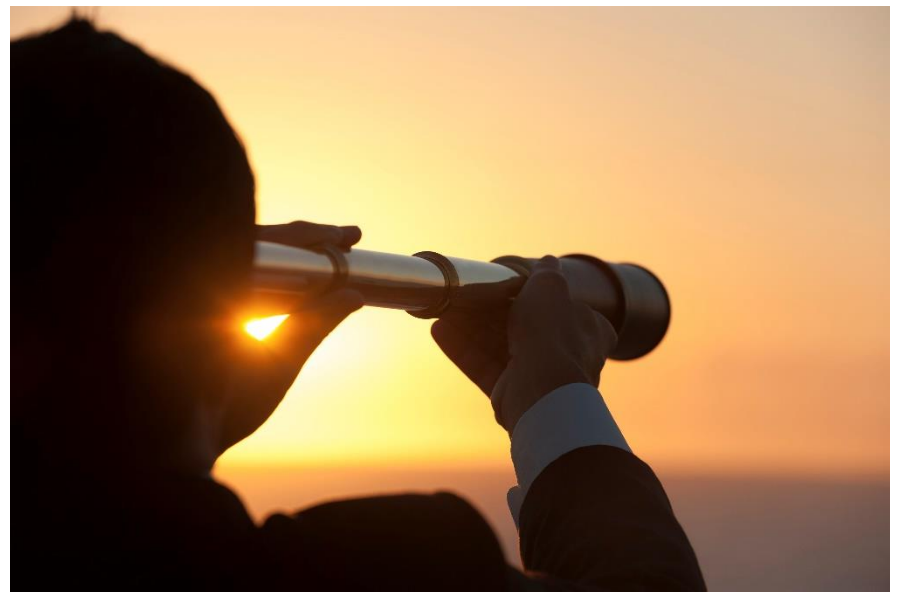
HONOURS PROGRAMME HP-SR in Information Technology

On how to train a machine to recognize these are the same animal

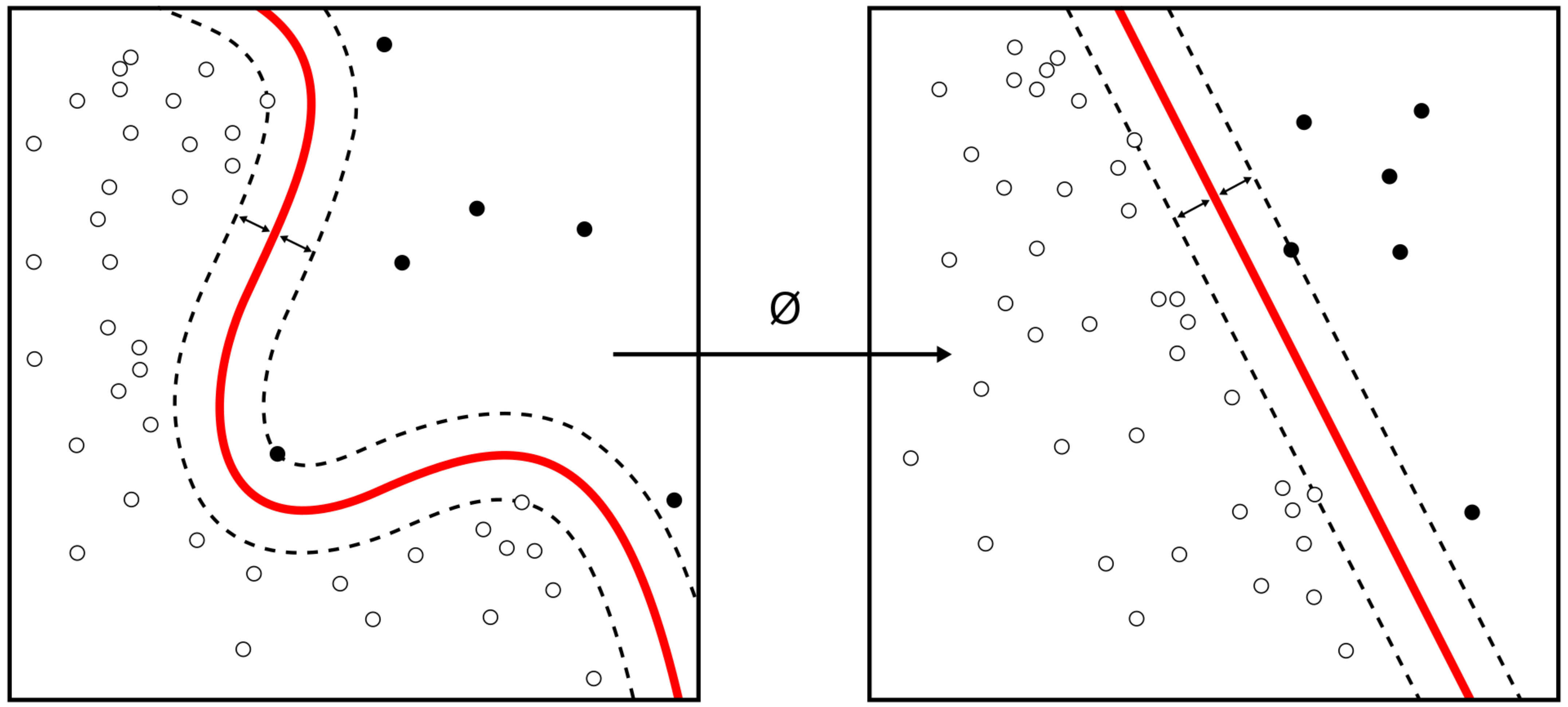




- With the least amount of samples
- Being able to generalize among different animals
- Being able to recognize unseen animals

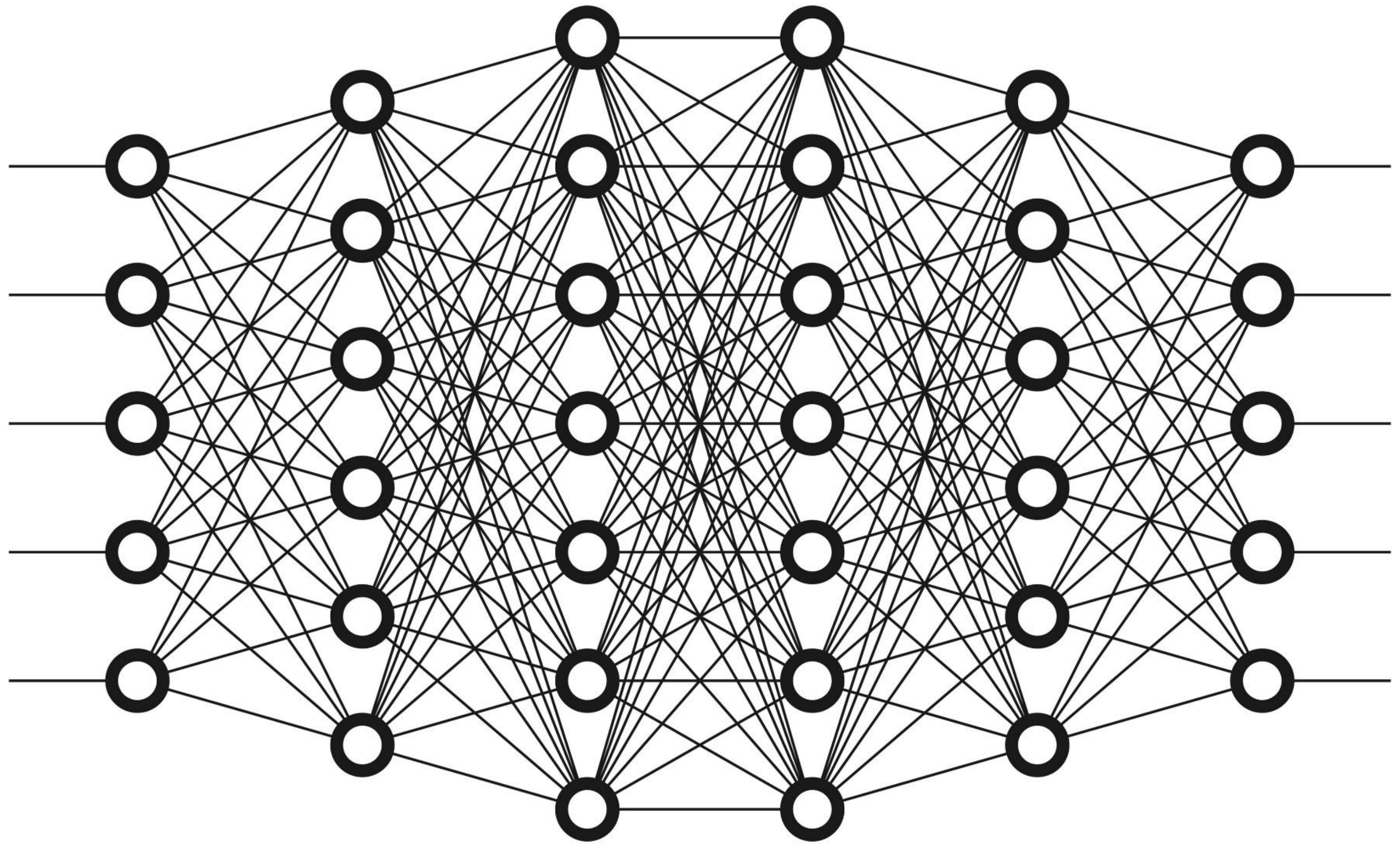


Machine Learning



- A subfield of Artificial Intelligence
- Extracts patterns from data
- Reuses identified patterns to make predictions
- Being able to recognize unseen

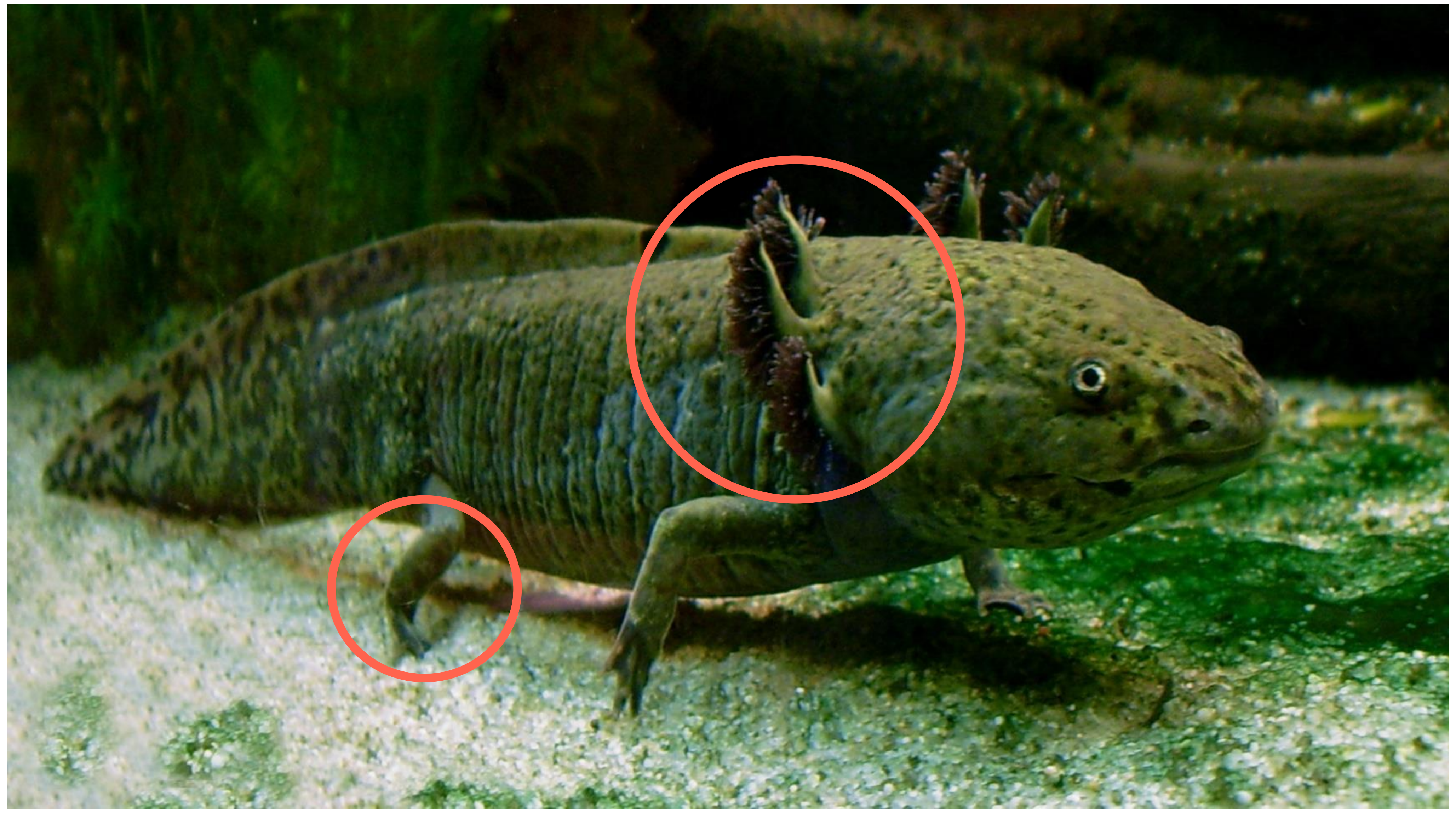
Deep Learning



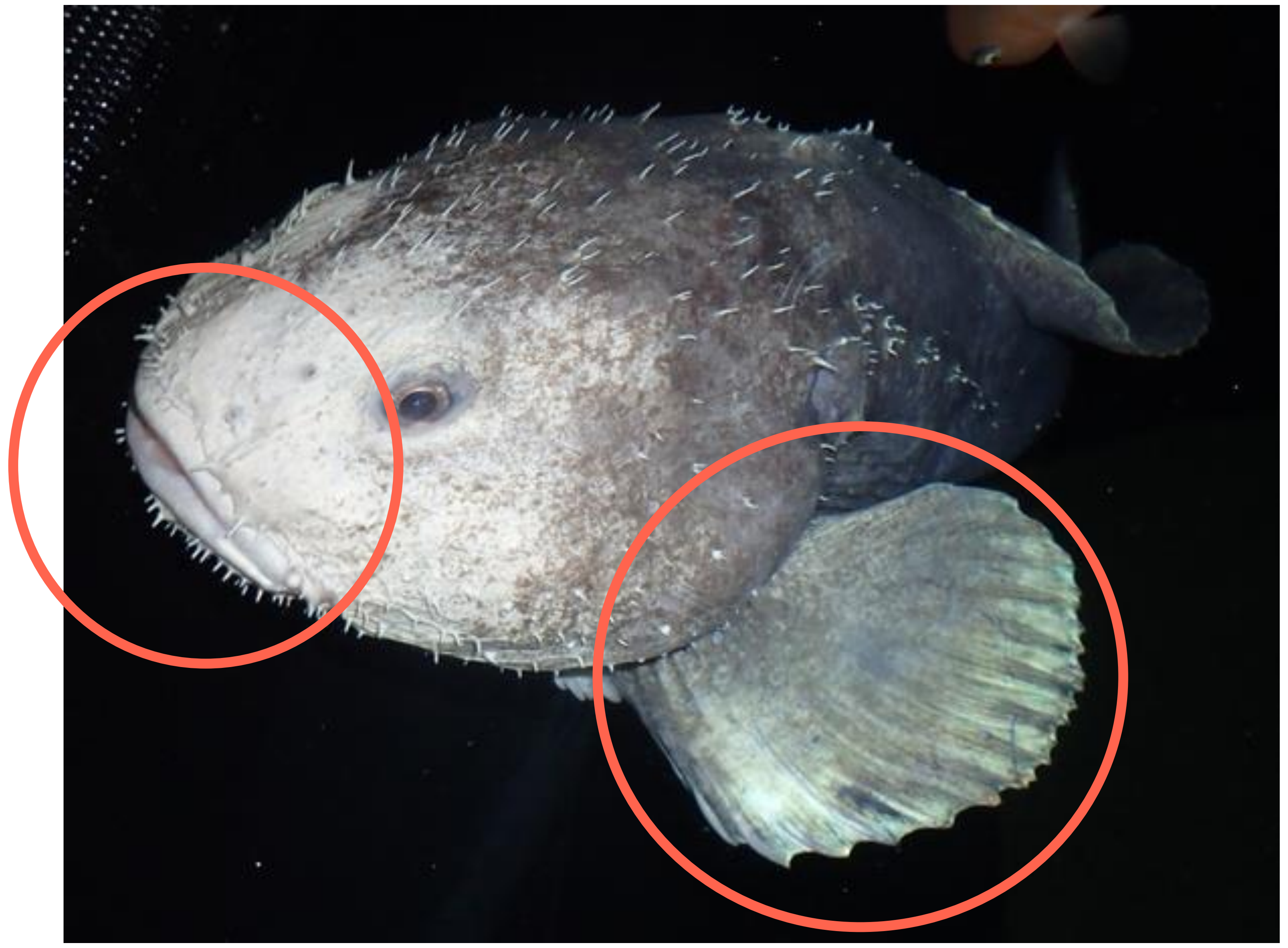
- Most popular branch of Machine Learning
- Deep neural networks learn feature extraction
  - Pros: No domain knowledge is required
  - Cons: Many parameters need to be updated
    - Incredibly data-hungry process

Classification

Axolotl



Blobfish

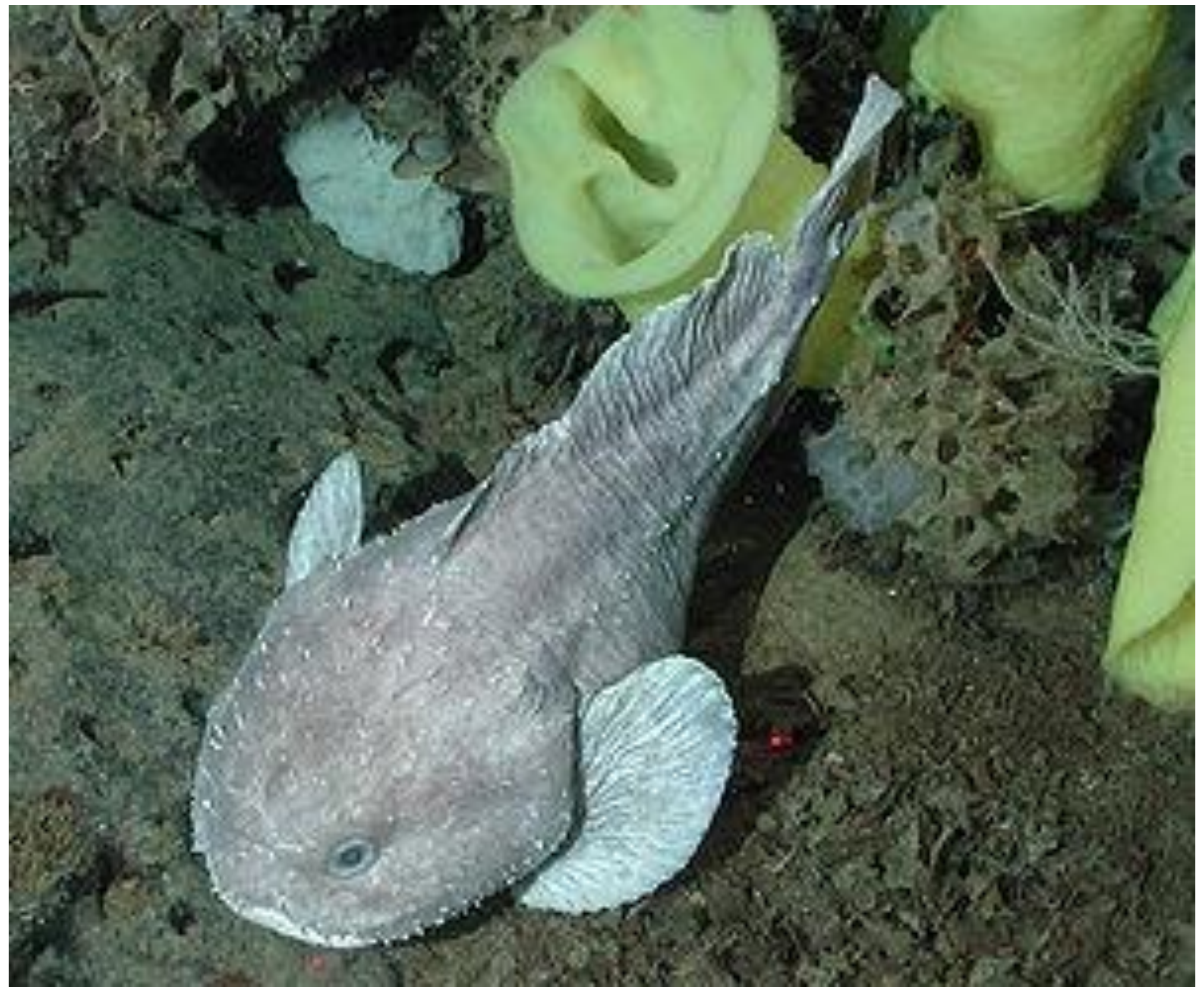


- This is a training set with just two images, each one with its own label

Classification

Axolotl

Blobfish



- After training, we are asked to label these new samples
- We just outperformed Deep Learning!

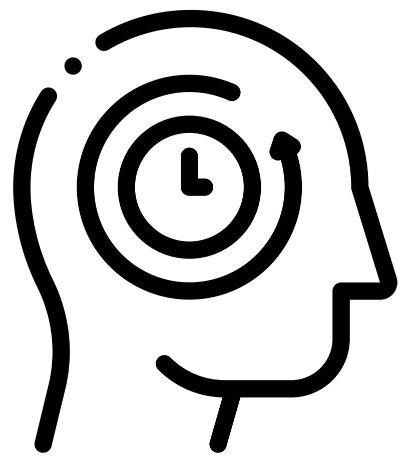
Few-Shot Learning

- The previous is an example of few-shot learning, where Deep Learning struggles
- Models are challenged to learn tasks with just a few training samples
    - Incredibly difficult, overfitting is insidious
    - Humans can easily deal with the problem

Our motivations

- Techniques dealing with these problems are currently on the rise
- Right now, many application domains lack training data (e.g. clinical data)
  - Our desired model can transfer knowledge from domains with plenty of data
    - Machine learning techniques will be able to be employed in more domains
- The approaches we work on may shed light on the way our brain works
  - Might play a crucial role in the discovery of Artificial General Intelligence (AGI)

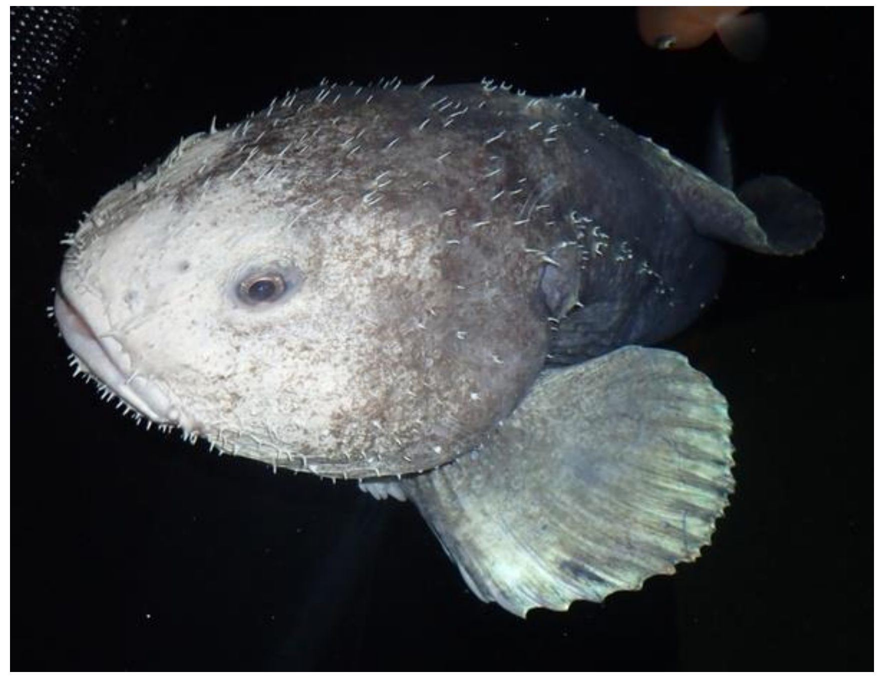
Experience

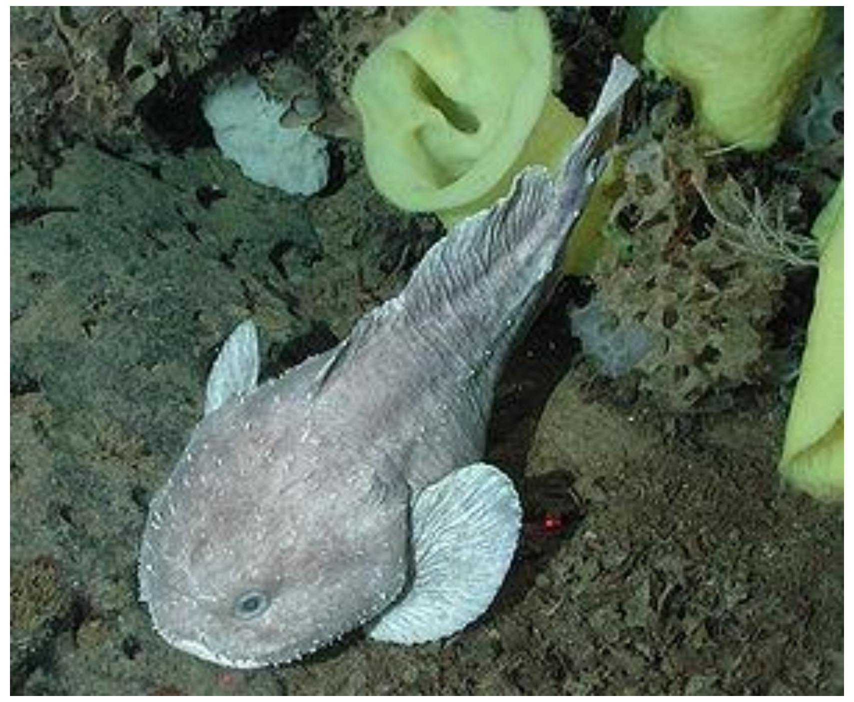


- Humans base their behavior and learning on their experience
- Knowledge obtained from experience on past tasks is reused to learn new tasks sharing some similarities
  - We can transfer relevant features from one ask to another, i.e., how to represent the problem
  - We can transfer the knowledge on how to solve the task, i.e., the “problem solving” strategy
- A task D in the experience can be modeled as samples from an unknown distribution over tasks $p(D)$

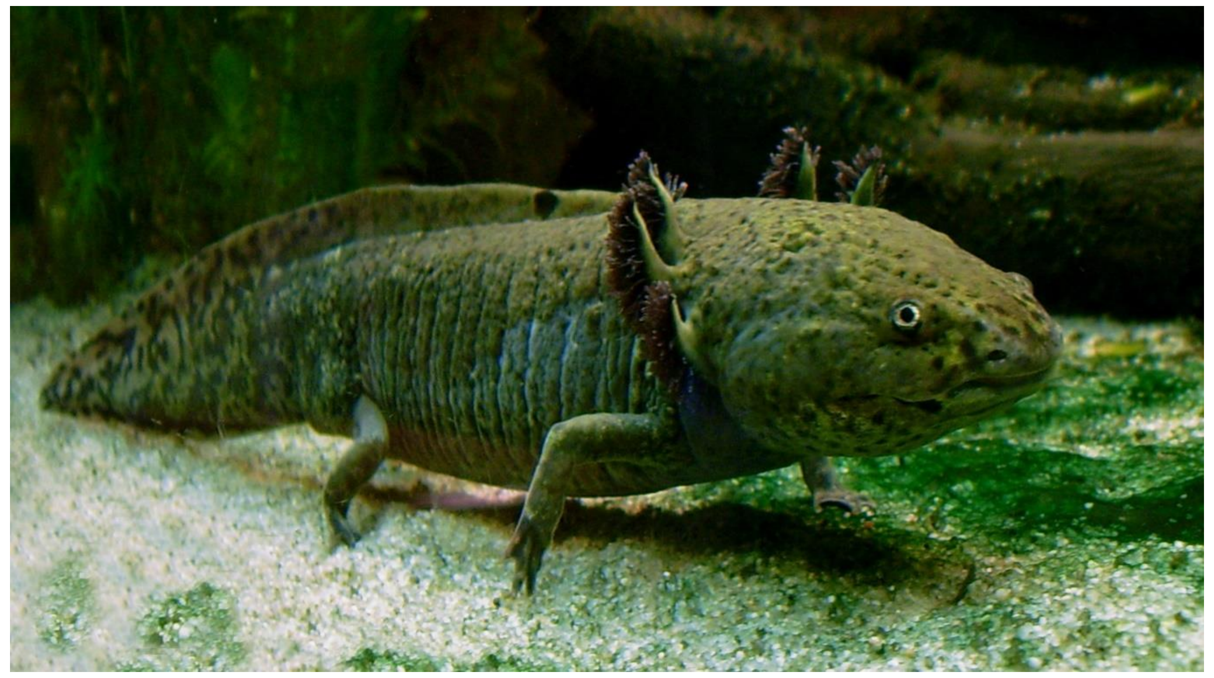
Learning

Blobfish



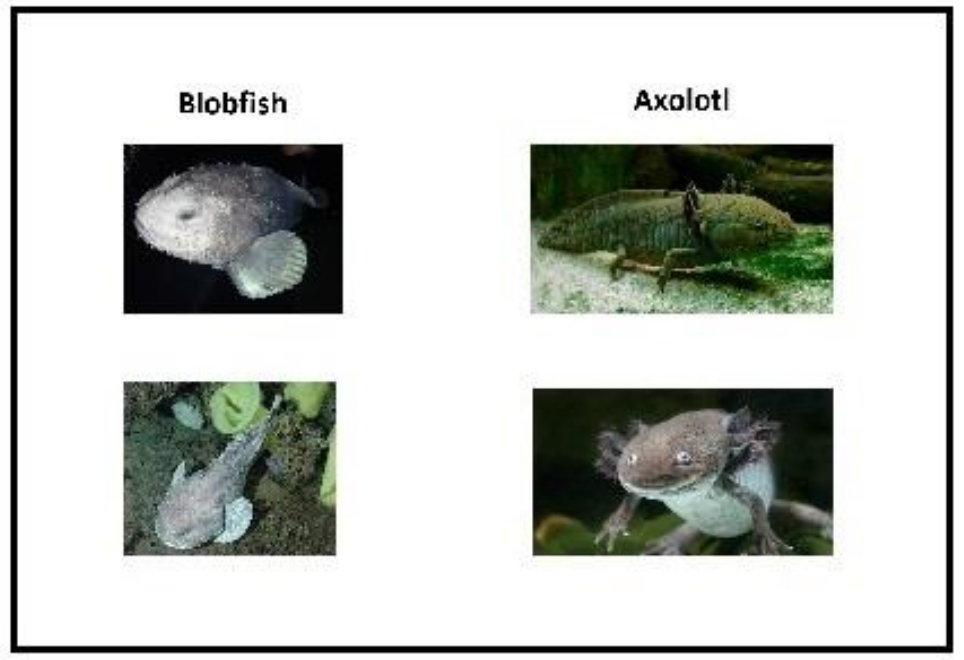


Axolotl

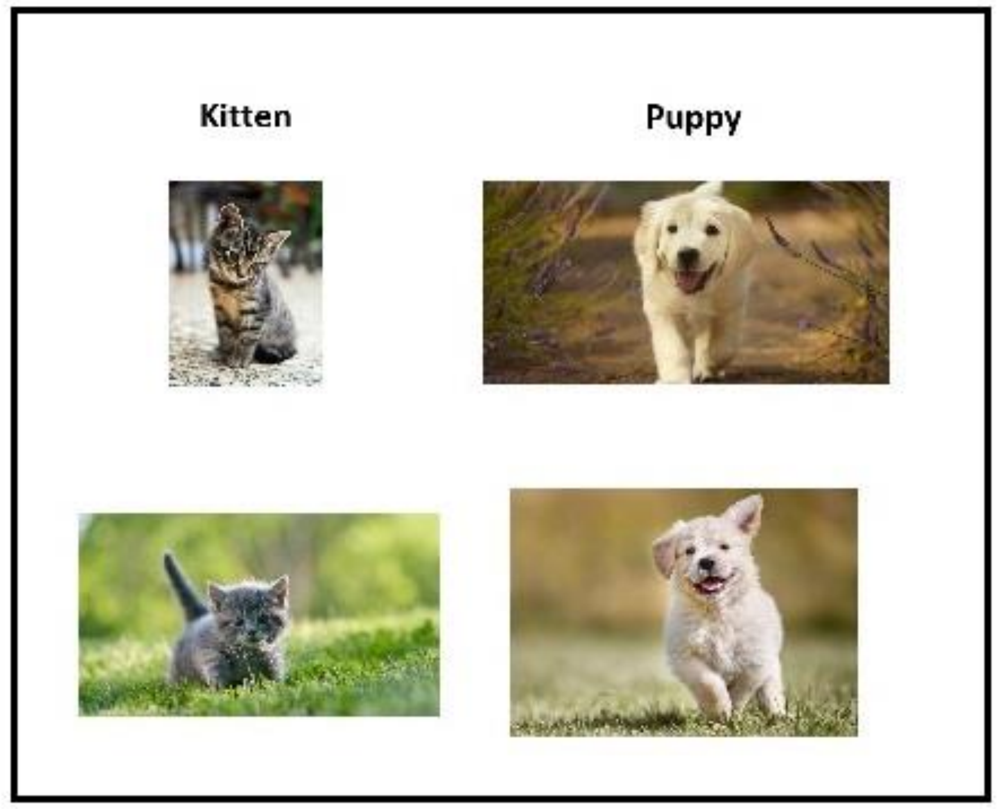


Meta-Learning

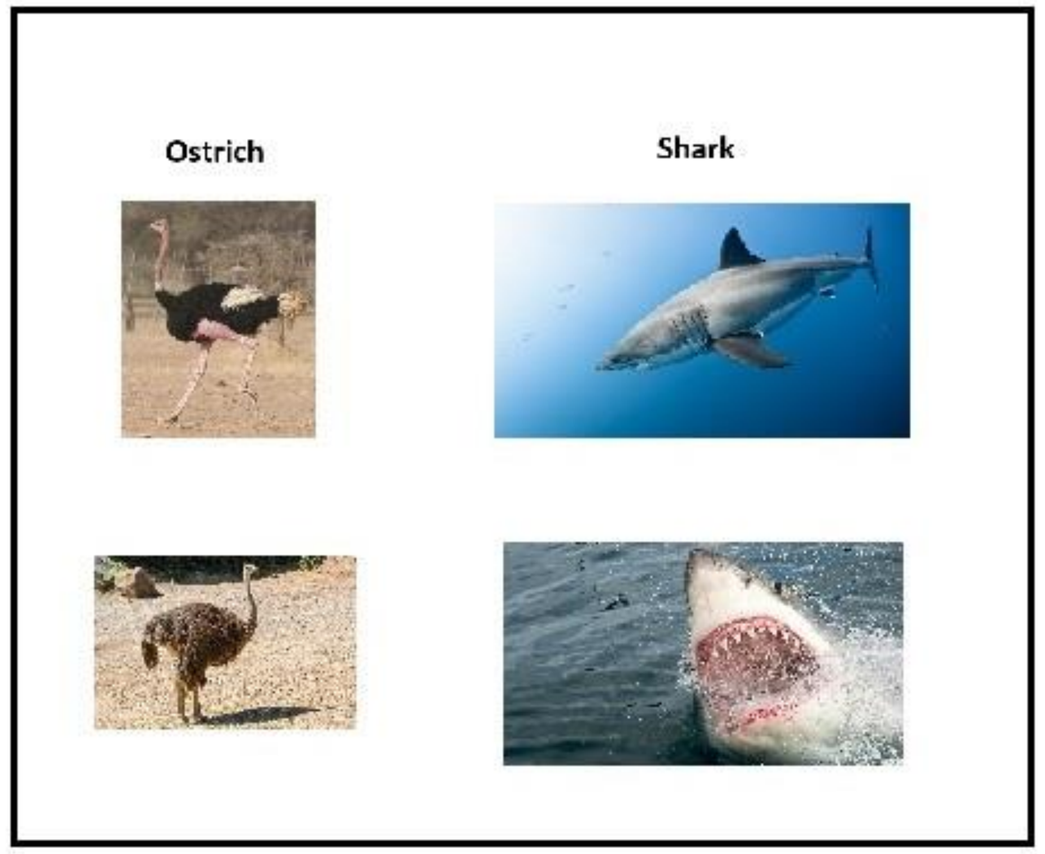
Task 1



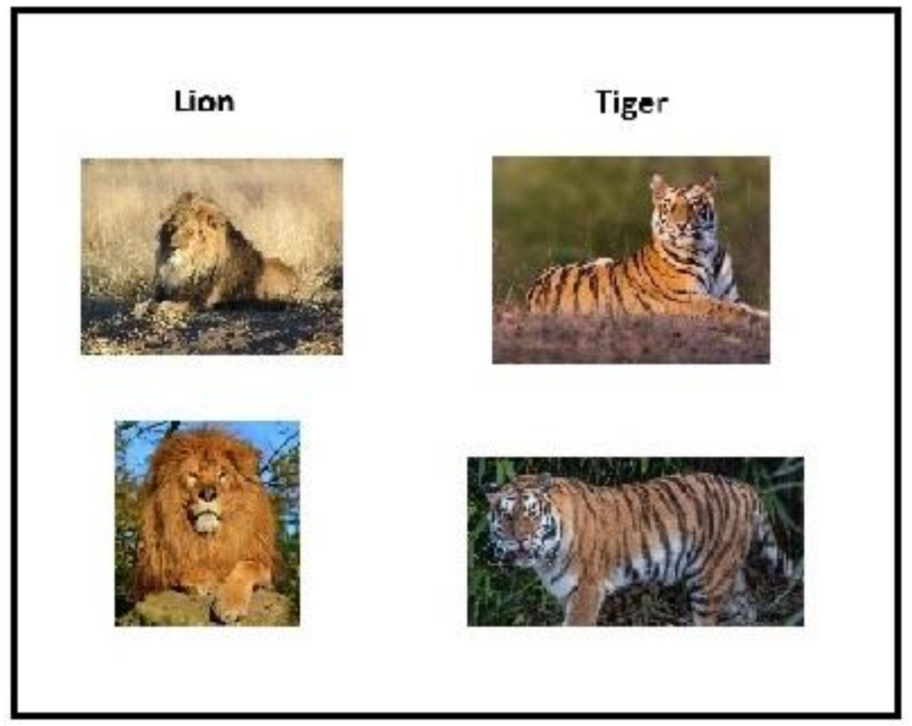
Task 2



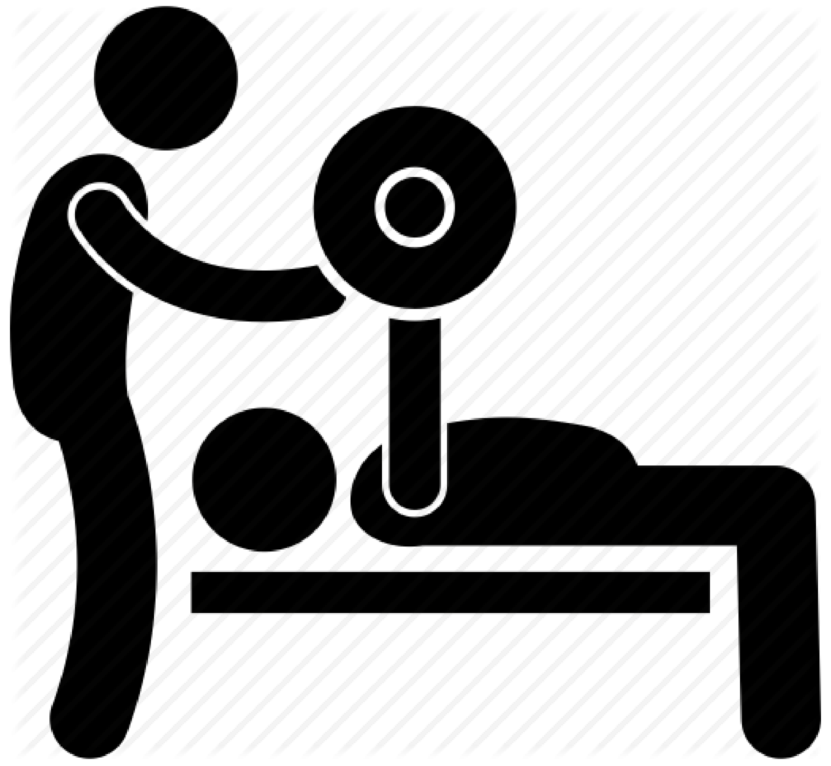
Task 3



Task N

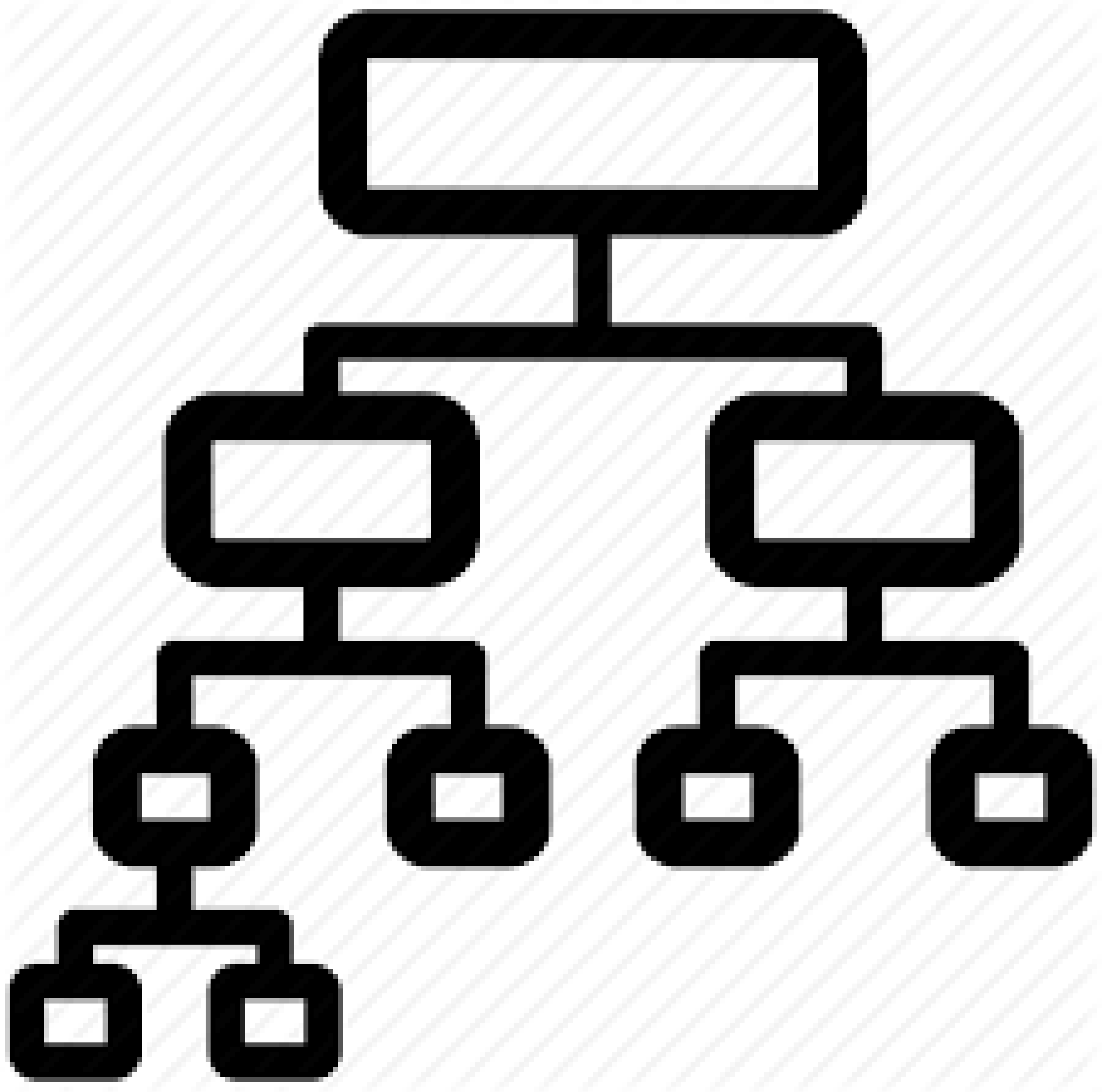


Meta-Learning



- The goal of Meta-Learning is "*Learning to learn*"
- We have a learner and a meta-learner
  - The meta-learner learns how to train the learner through a **collection of tasks**
  - The meta-learner objective is to **generalize** the knowledge on **unseen tasks**
  - As a result, the learner is trained on new tasks more effectively
- Leverages Deep Learning

Taxonomy



- **Metric-based**: leverages a kernel function as a similarity measure among datapoints
- **Model-based**: focuses on models that learn quickly thanks to their structure or the use of a powerful meta-learner
- **Optimization-based**: looks for an optimal parameter initialization for the learner

↑ We focus here

MAML

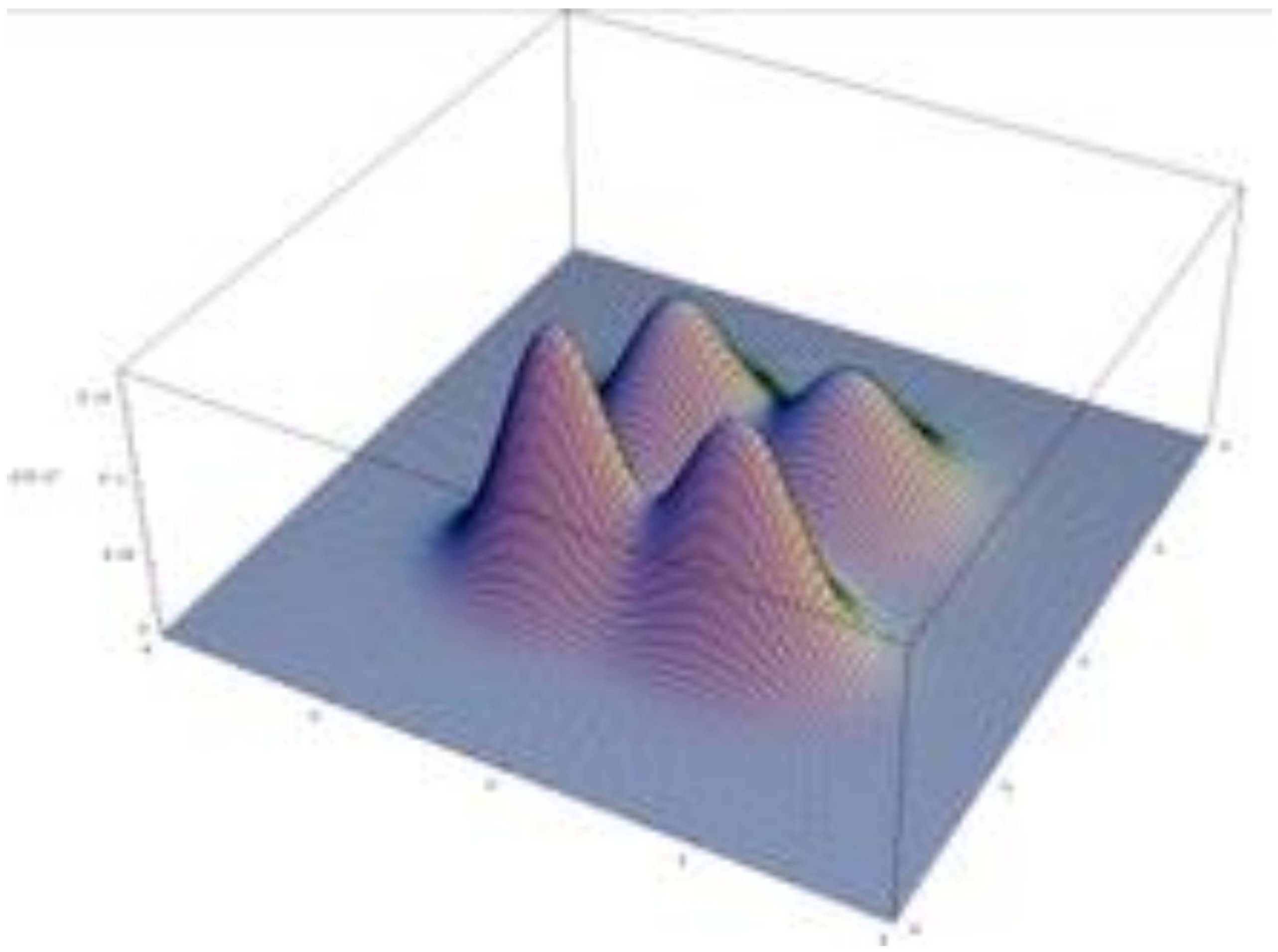
Require: $p(\mathcal{T})$: distribution over tasks
Require: α, β: step size hyperparameters

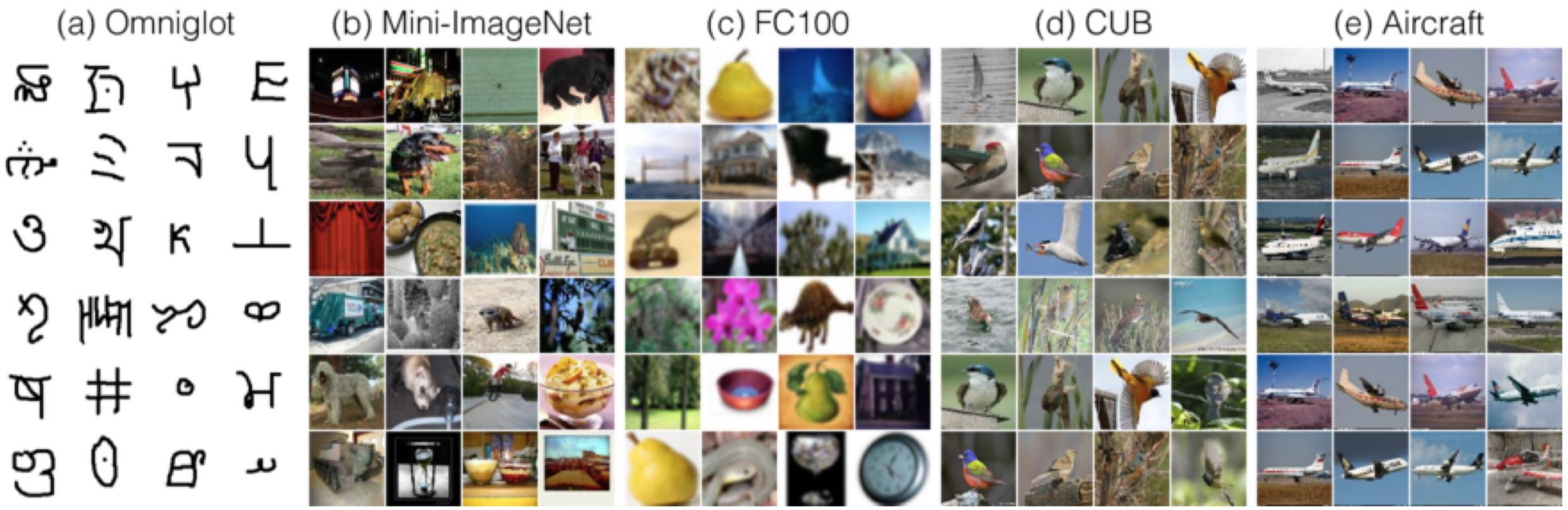
```
randomly initialize θ
while not done do
    Sample batch of tasks T_i ~ p(T)
    for all T_i do
        Evaluate ∇_θ L_{T_i}(f_θ) with respect to K examples
        Compute adapted parameters with gradient descent: θ'_i = θ − α∇_θ L_{T_i}(f_θ)
    end for
    Update θ ← θ − β∇_θ Σ_{T_i~p(T)} L_{T_i}(f_{θ'_i})
end while
```

- Applies to a generic learner f_θ
- Only requirement for f_θ is to be trainable via gradient descent
- Finds a single parameter initialization for f_θ
- The algorithm features an inner and outer loop

[Finn et al. 2018]

Multimodal MAML

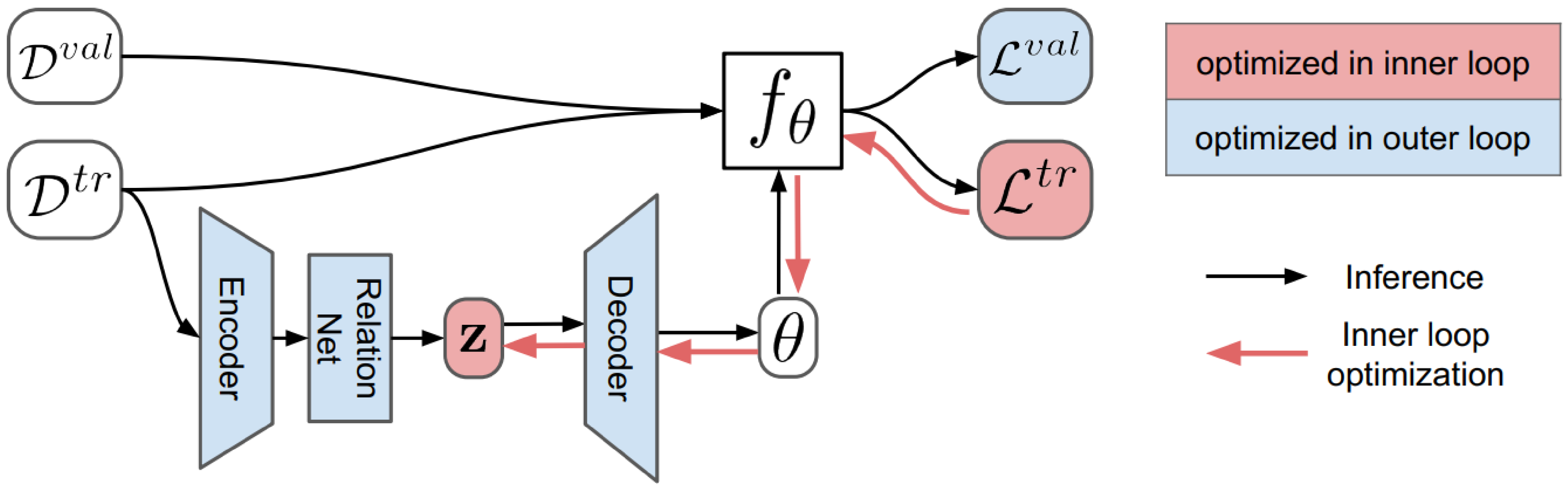




- Extends MAML
- Operates in multimodal task distributions
- Recognizes the mode of a task
- Finds different initializations for different modes
- Limited knowledge sharing

[Vuorio et al. 2019]

LEO

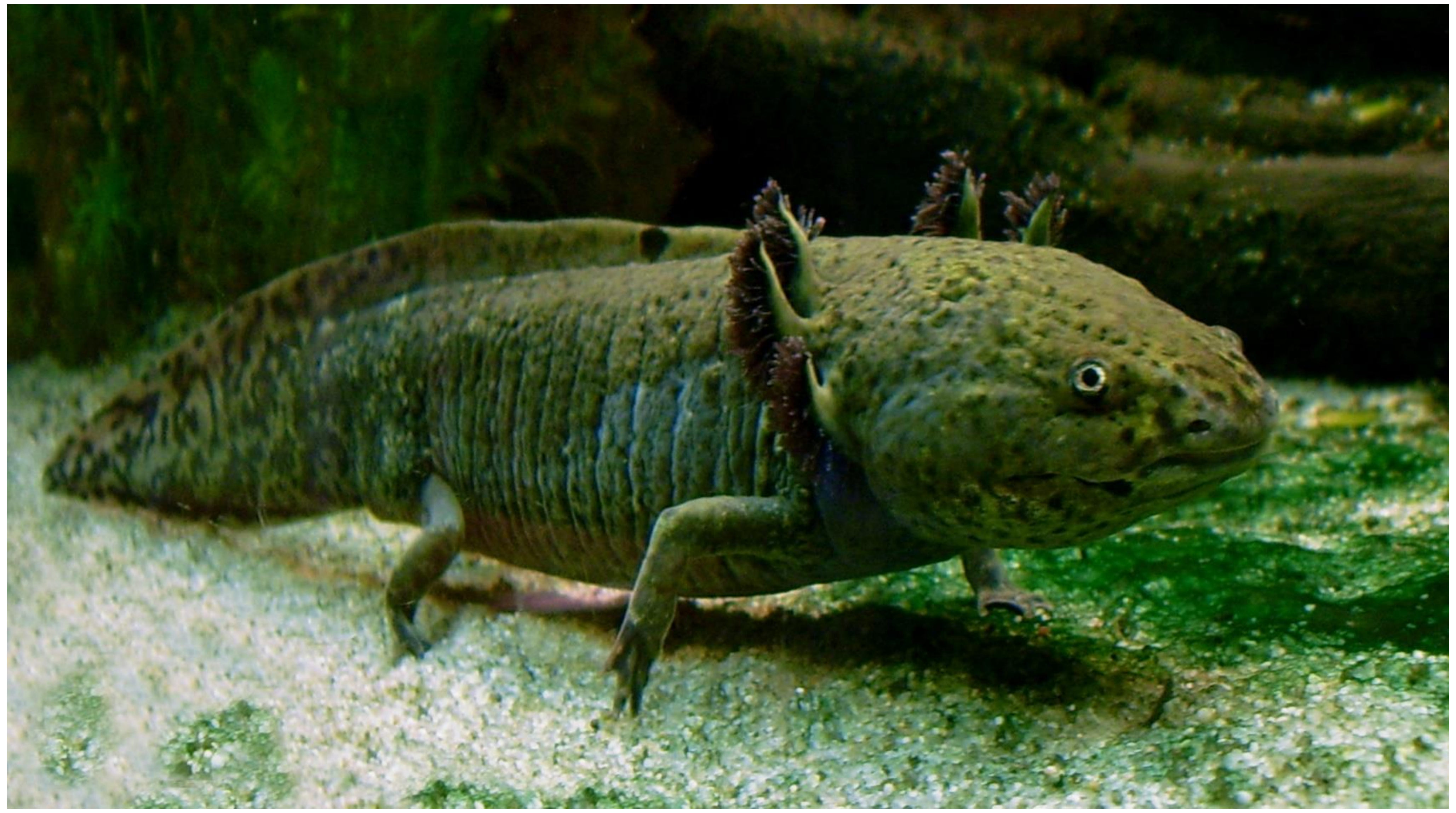


- Like MAML, a meta-learner trains the learner f_θ through a double loop
- Enriches f_θ with encoder and decoder
- Encodes the task as a latent representation
- Derives parameters from the latent representation
- Performs gradient descent in the latent space

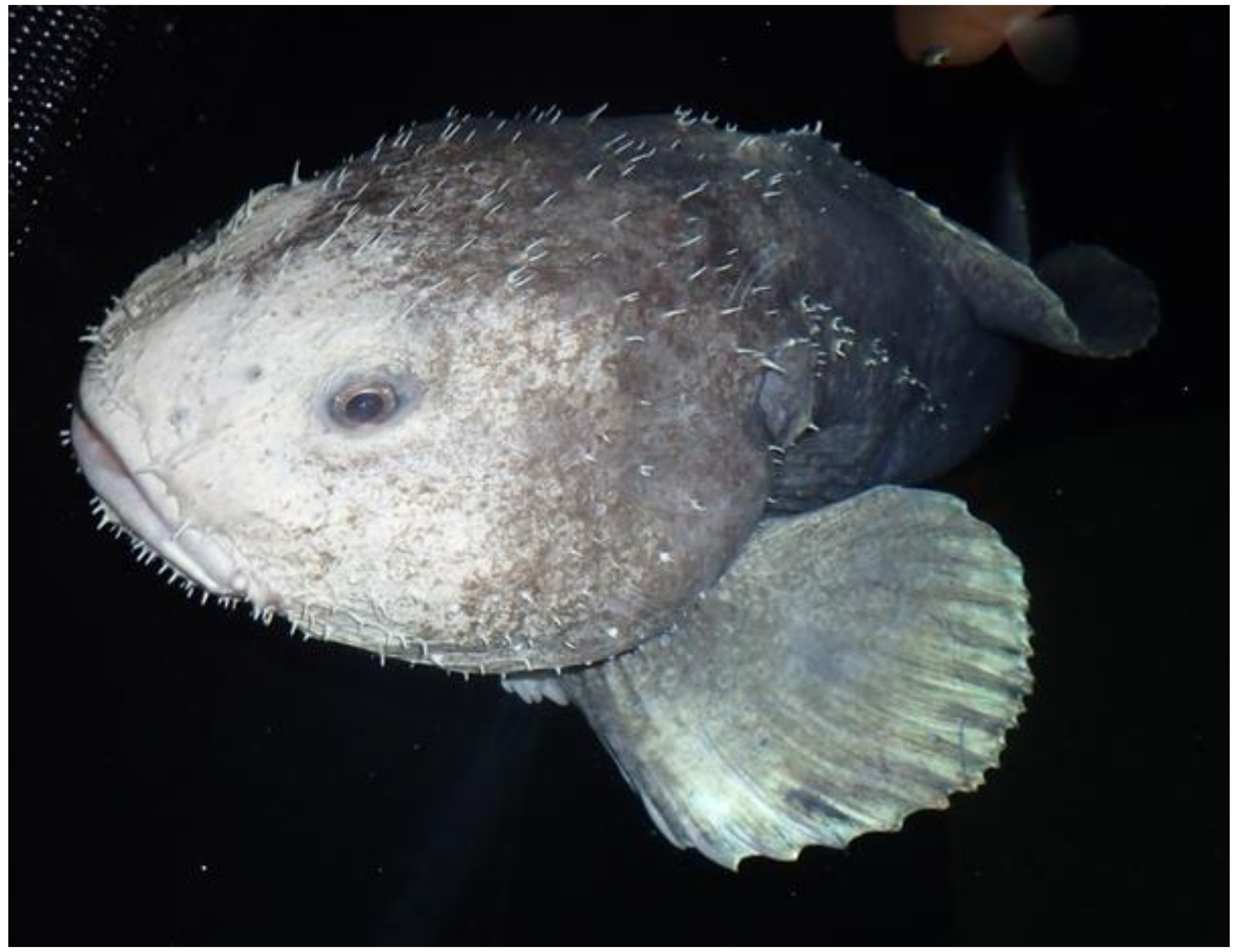
[Rusu et al. 2019]

One-Shot Classification

Axolotl



Blobfish

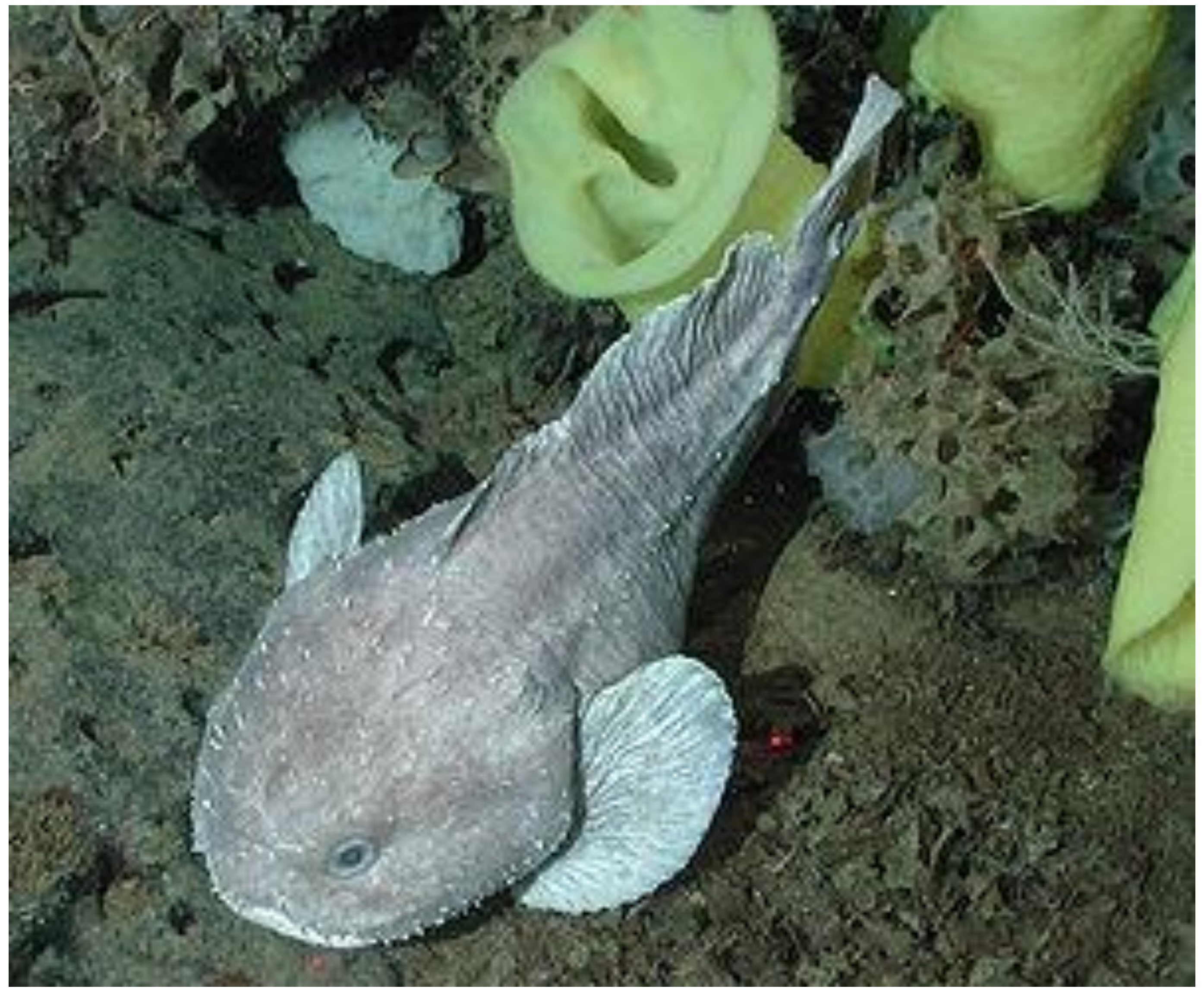


- Let’s reconsider the task from before

One-Shot Classification

Axolotl

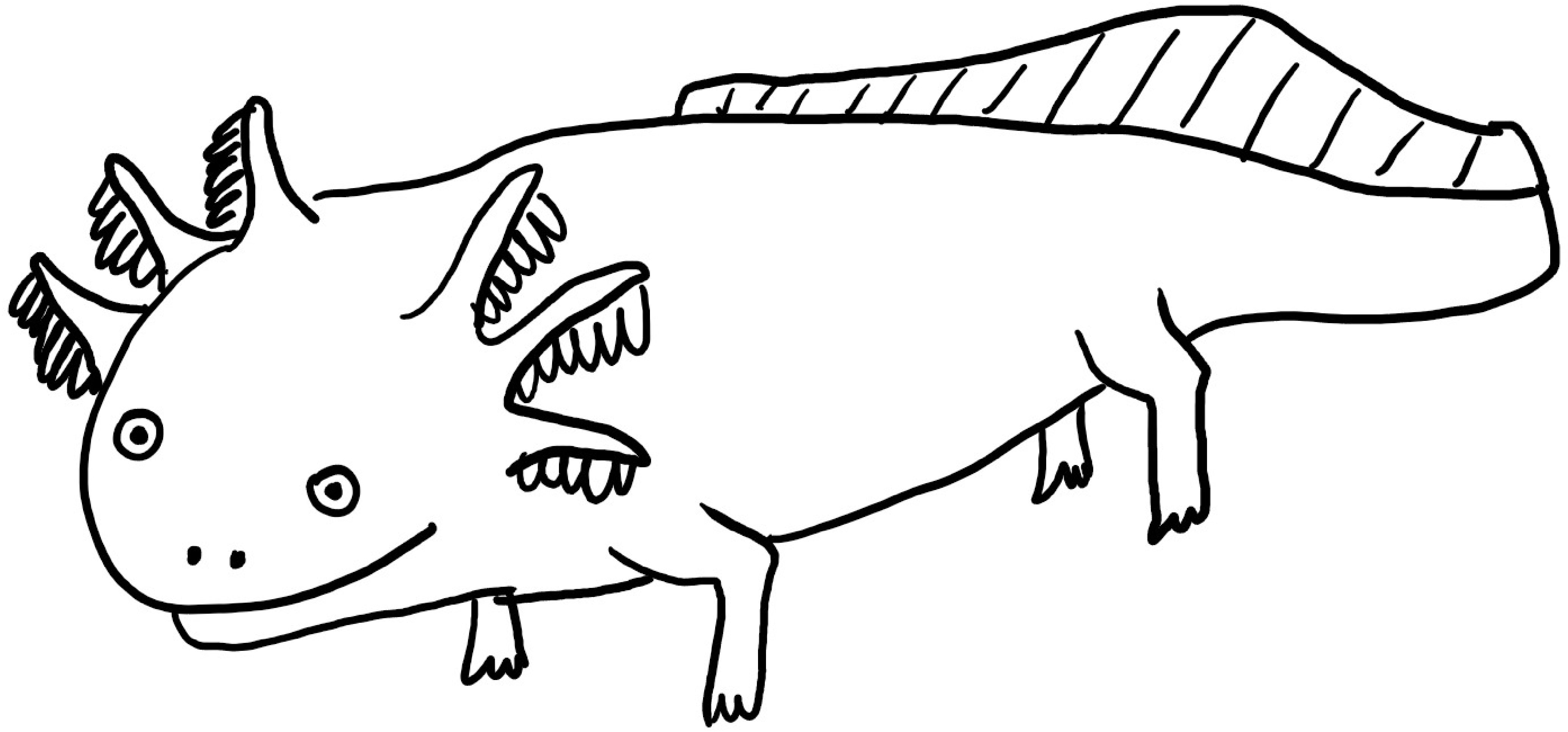
Blobfish



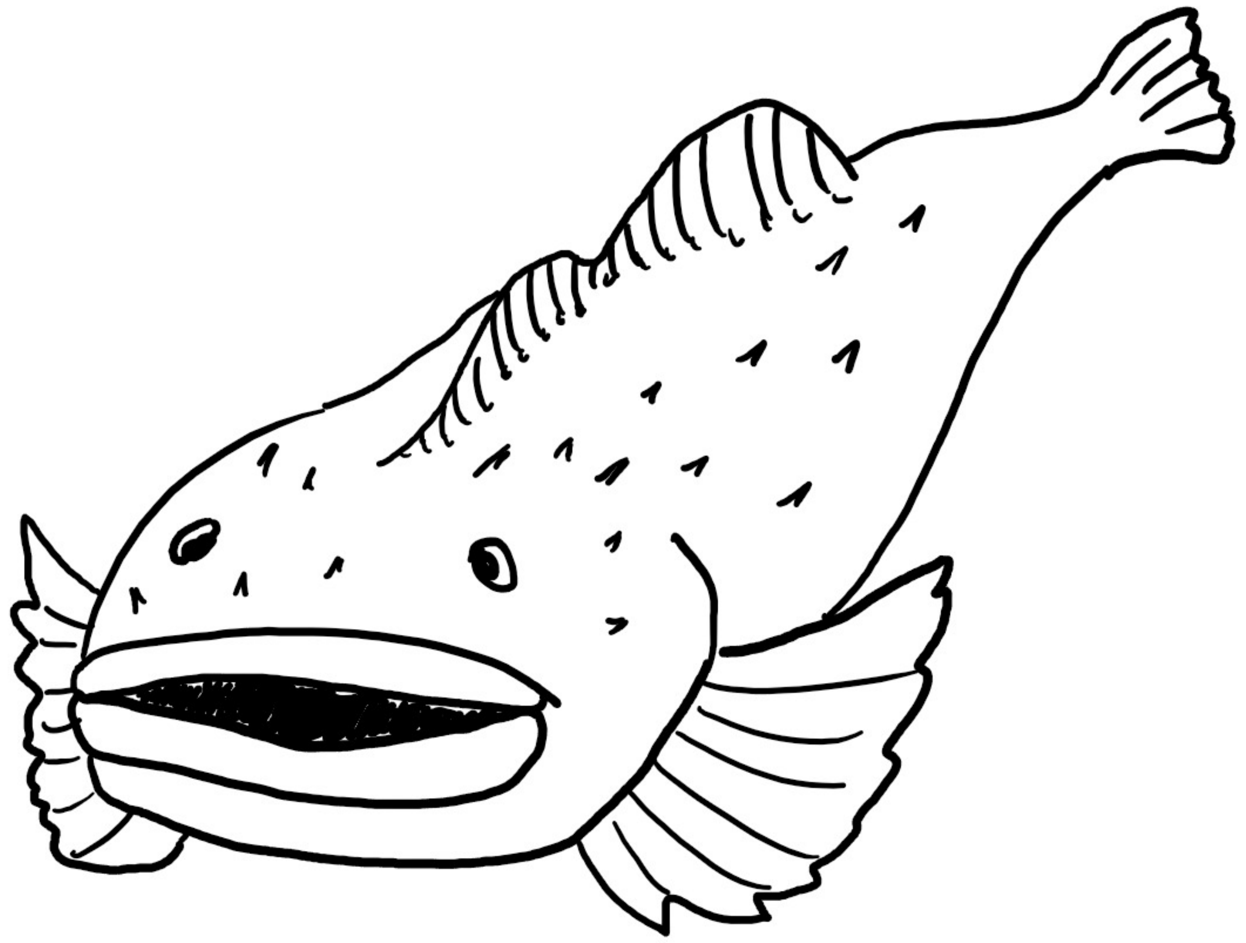
- Let's reconsider the task from before
- Meta-Learning can do this

One-Shot Classification

Axolotl



Blobfish



- Let's try with two other images
- We are smarter than Meta-Learning!

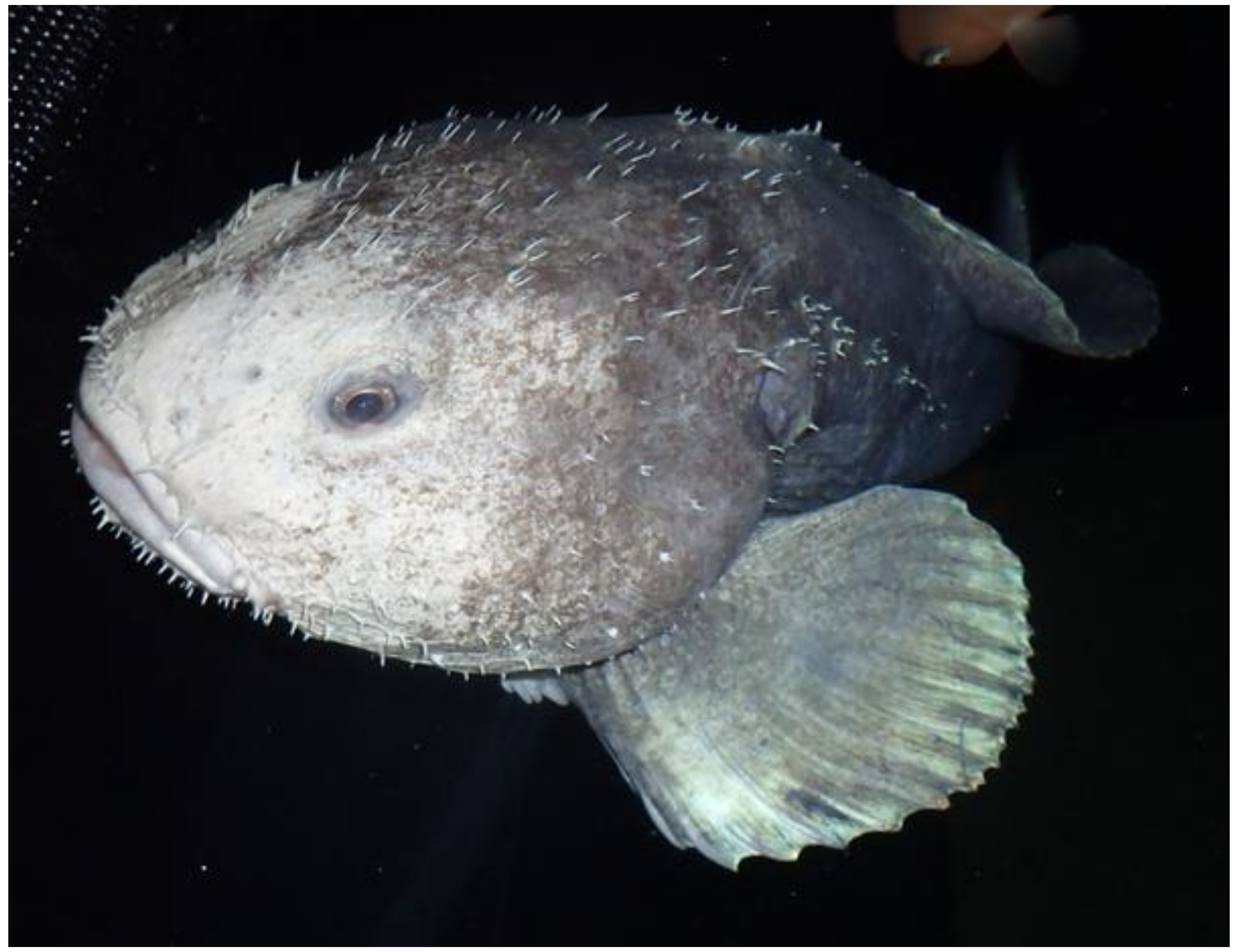
Heterogeneous Domains

- The previous is an example of cross-domain learning
- Cross-domain learning deals with data coming from heterogeneous domains
  - Examples of heterogeneous domains are different source cameras or different light conditions
- Humans vastly outperform current meta-learning techniques in heterogeneous domains, so we believe there is much room for improvement in this case

Beyond heterogeneous domains

Dragon

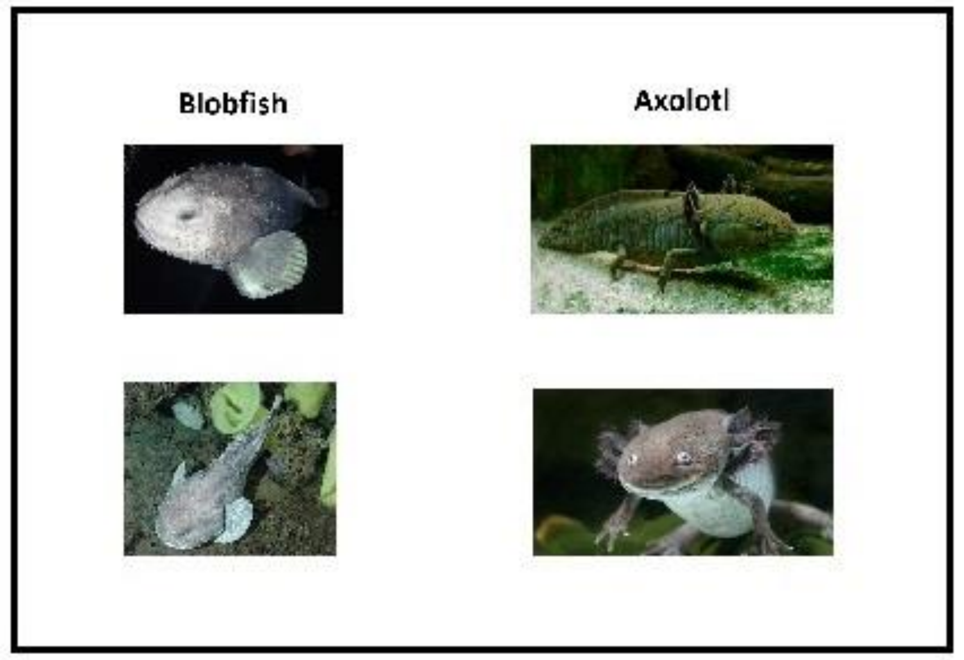
Blobfish



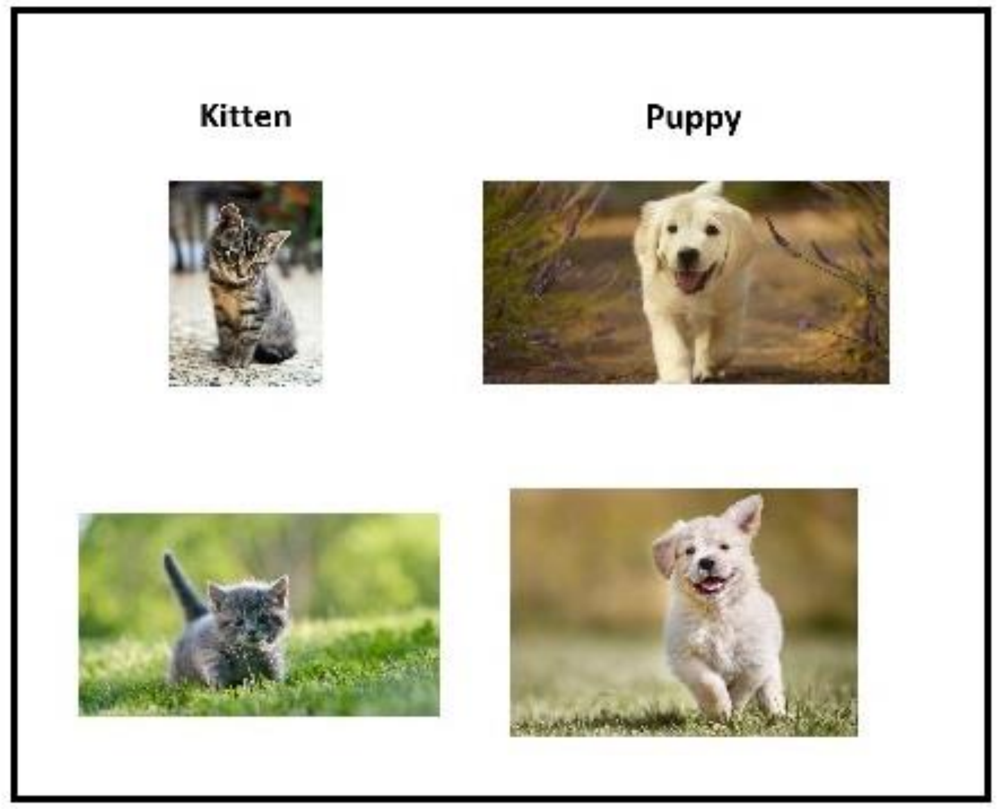
- This is another new task that may be difficult to learn for Meta-Learning

Beyond heterogeneous domains

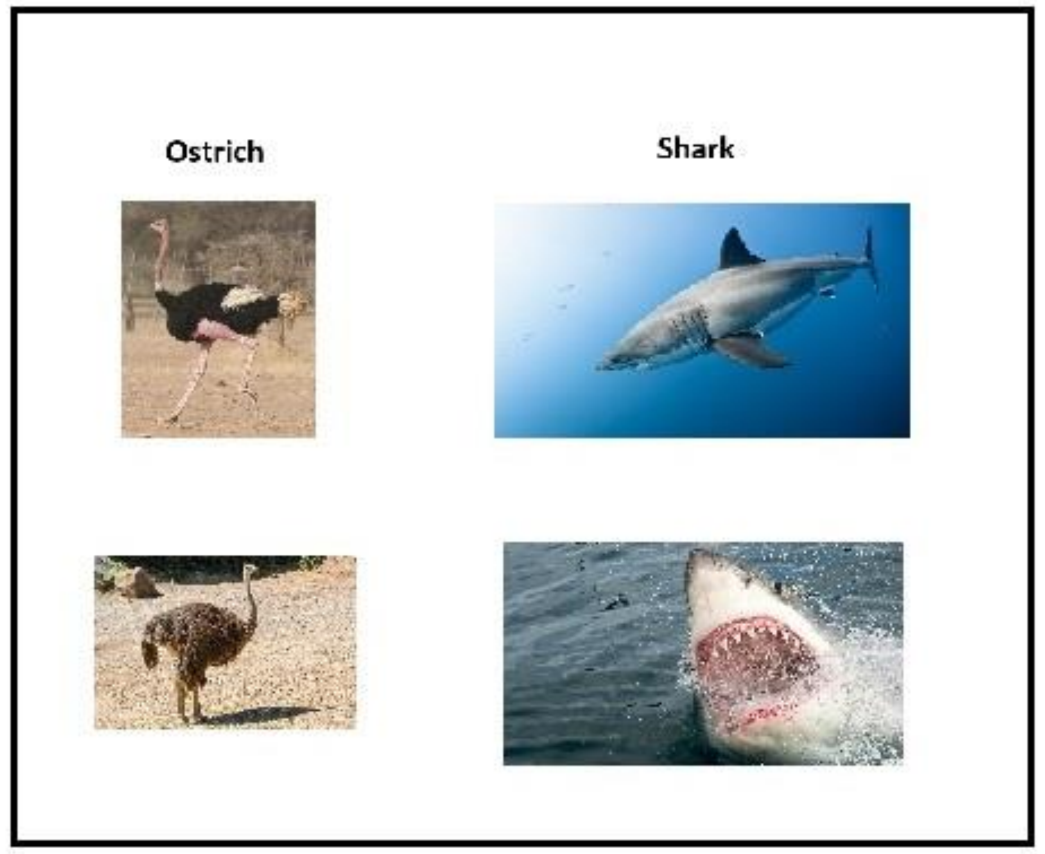
Task 1



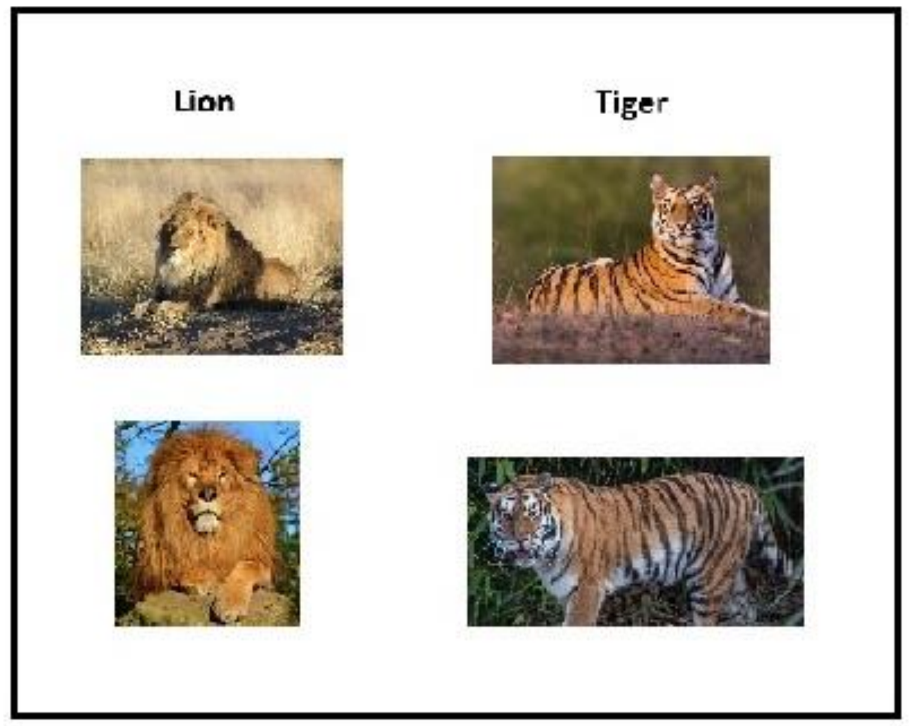
Task 2



Task 3

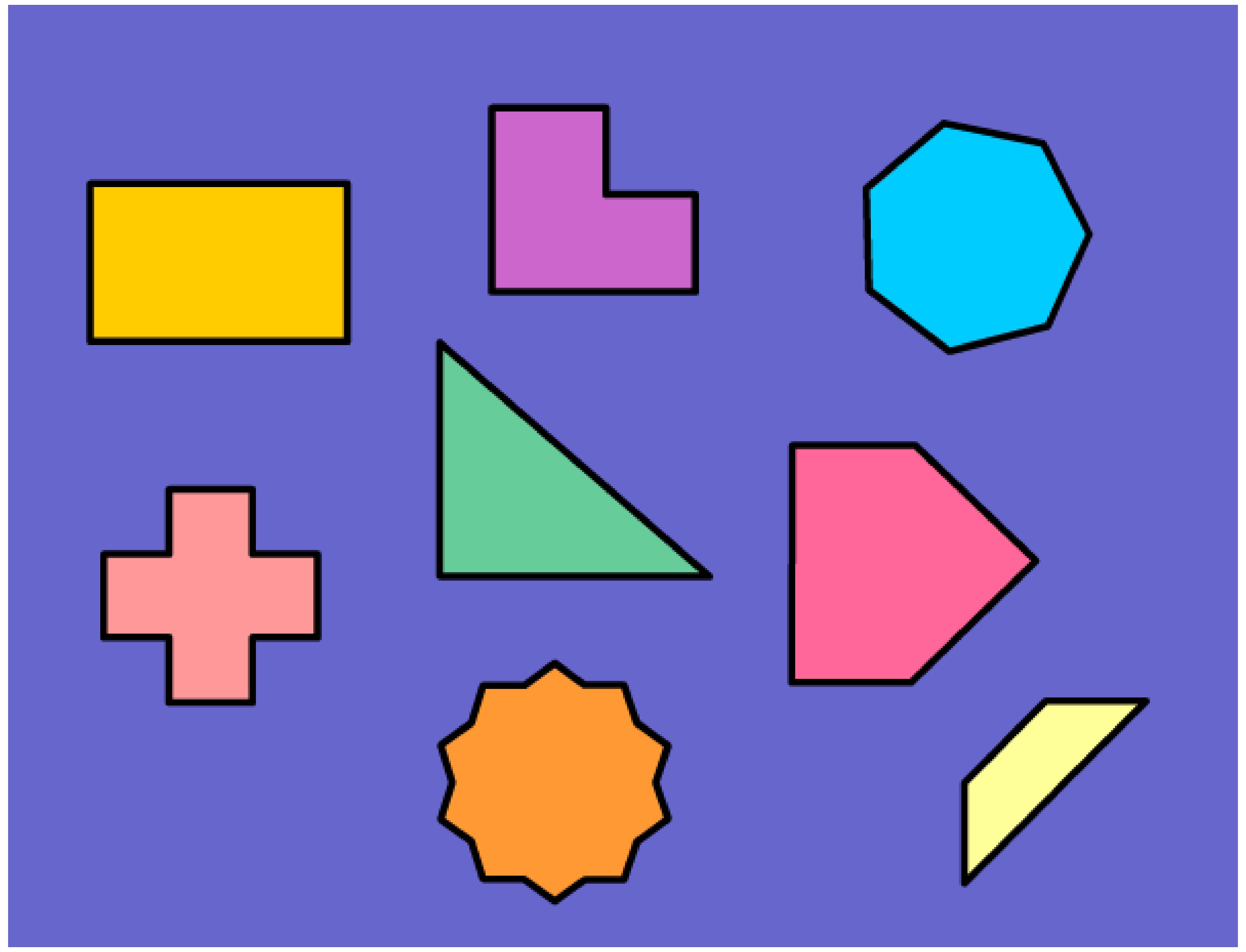


Task N



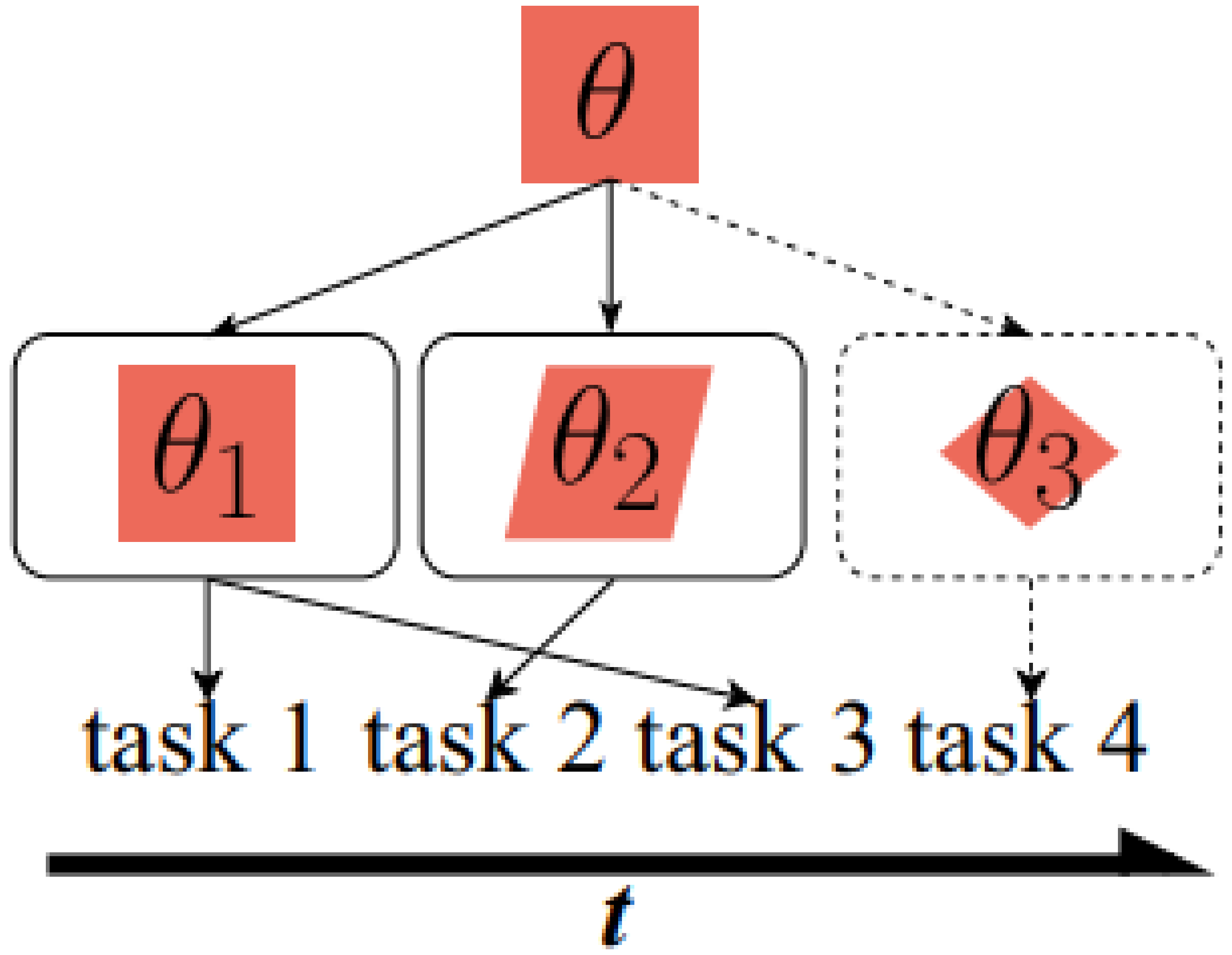
- Fictional animals like dragons are nowhere to be seen in the dataset

Heterogeneous Tasks



- Domain heterogeneity is a particular case of heterogeneous tasks
- Heterogeneous tasks can vary in many aspects, like domain, number of classes, number of shots…
- The problem of heterogeneous tasks has been addressed by some approaches in the literature

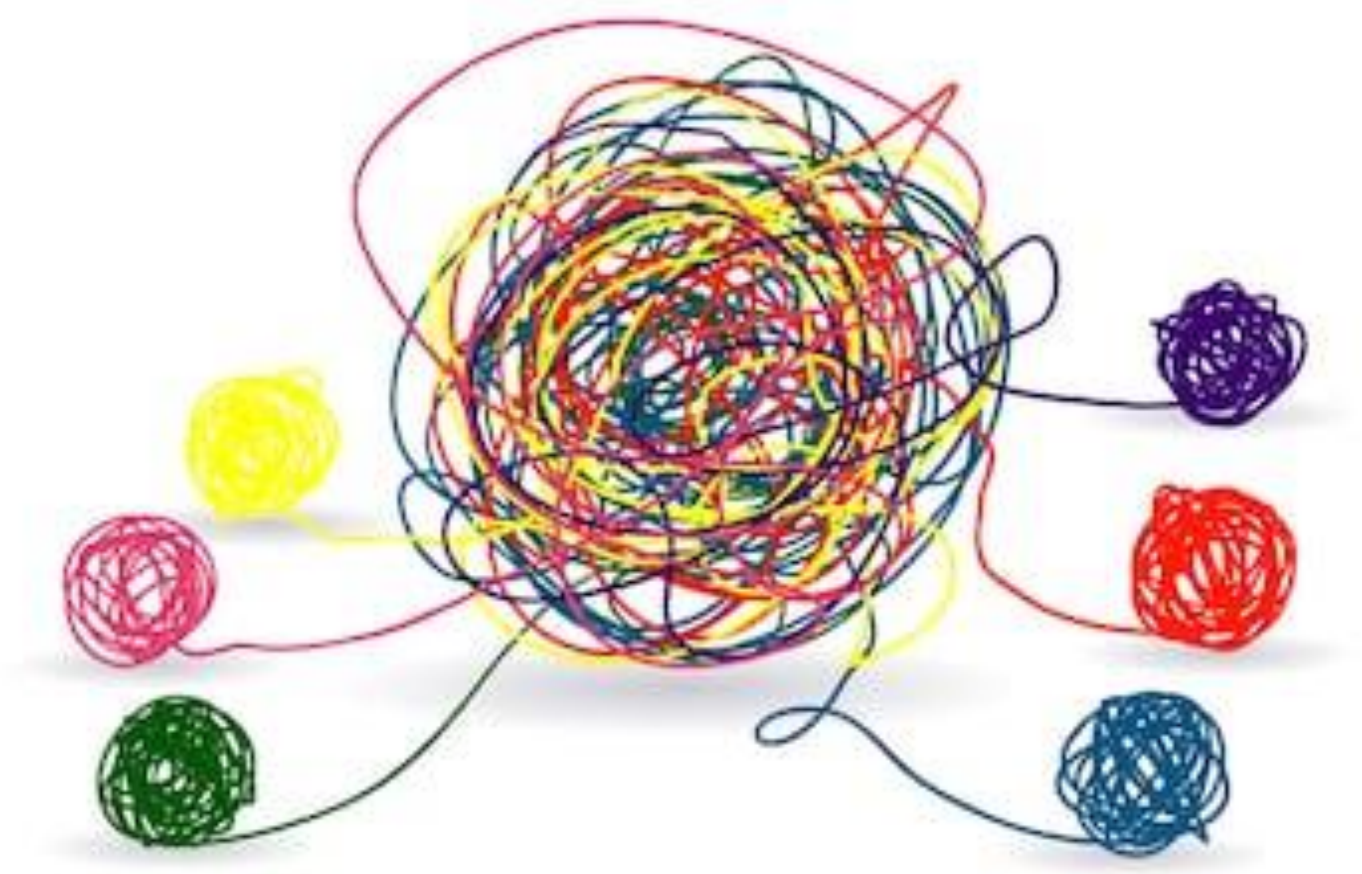
HSML



- Arranges task in a hierarchical structure
- Hierarchical tree is built over time
- Shares knowledge among tasks from the same hierarchy
  - Knowledge can be initial parameters (not necessarily)
- Only one hierarchy may be limiting

[Yao et al. 2019]

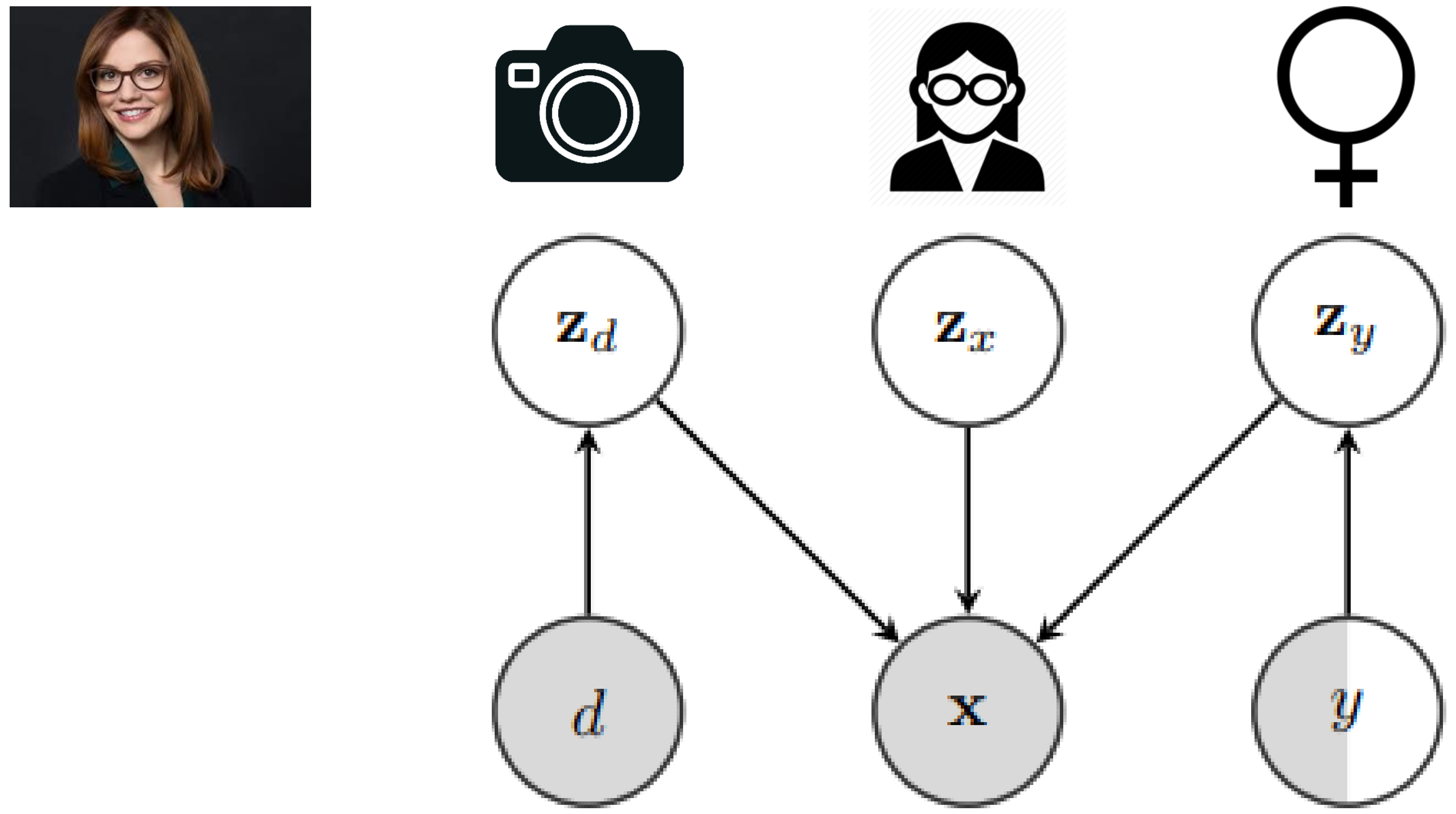
Our approach



- We focus on heterogeneous domains
- Some recent works argue that a meaningful embedding could prove useful
    - It can be obtained through a **disentangled representation**, capturing different independent properties of data in different units
        - Disentanglement leads to high interpretability and reusability
- We combine previous architectures with a disentangled embedding

Disentangled Faces

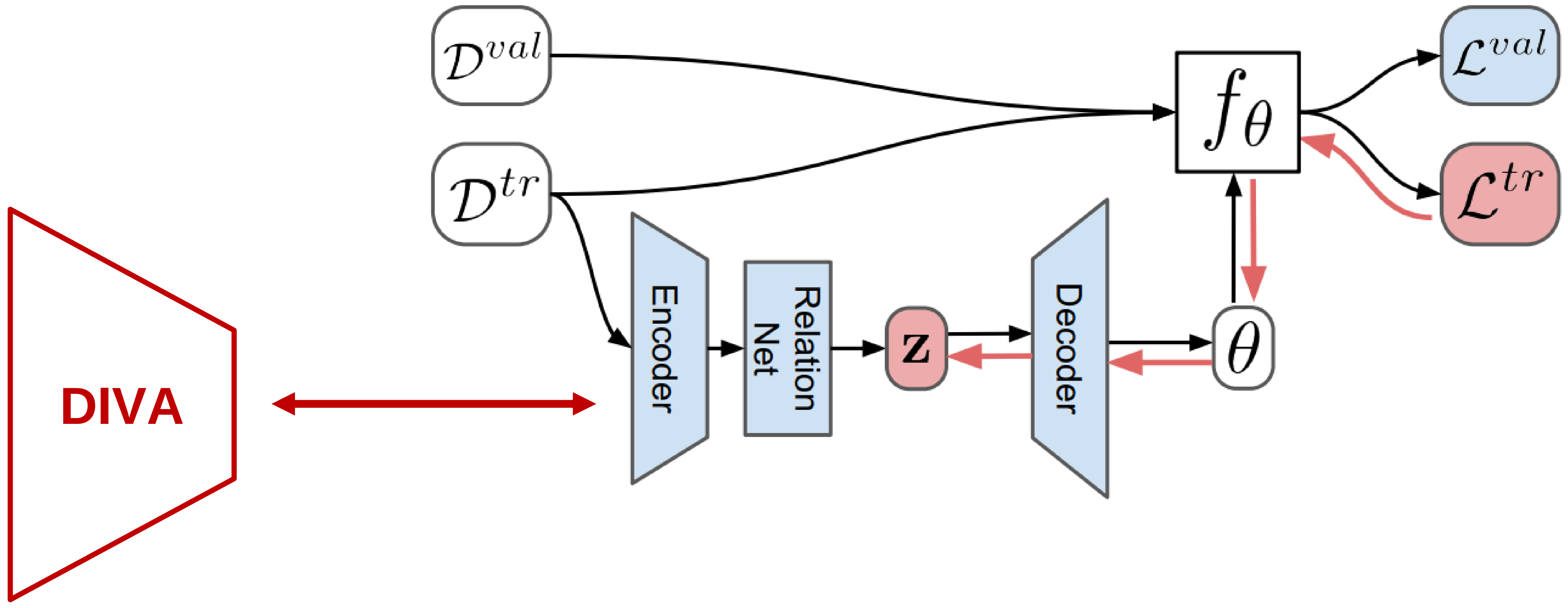
DIVA



- Domain Invariant Variational Autoencoder
- Encodes the input in a disentangled representation
- Can learn in a semi-supervised setting where only the domain label is provided
- The latent space is divided in 3 different subspaces
  - Domain, residual, and class
- Can generalize to unseen domains

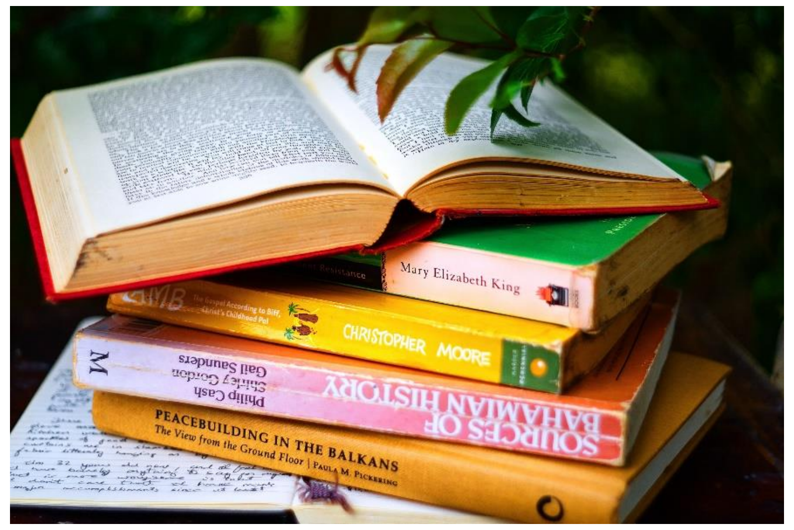
[Ilse et al. 2019]

Our idea



- We choose to combine DIVA with LEO
- Multiple reasons for this:
  - LEO also makes use of embeddings
  - Compressed LEO representation for gradient-descent

Schedule



1. Literature Review
2. Implementation
3. Observation
4. Experimentation
5. Paper

References

- Finn et al. Model-agnostic meta-learning for fast adaptation of deep networks, 2018
- Rusu et al. Meta-learning with latent embedding optimization , 2019
- Vuorio et al. Multimodal model-agnostic meta-learning via task-aware modulation , 2019
- Yao et al. Hierarchically structured meta-learning , 2019
- Ilse et al. Domain invariant variational autoencoders, 2019