Learning with Small Samples

Including zero-shot learning

Nour Karessli
DSR 2019

Structure

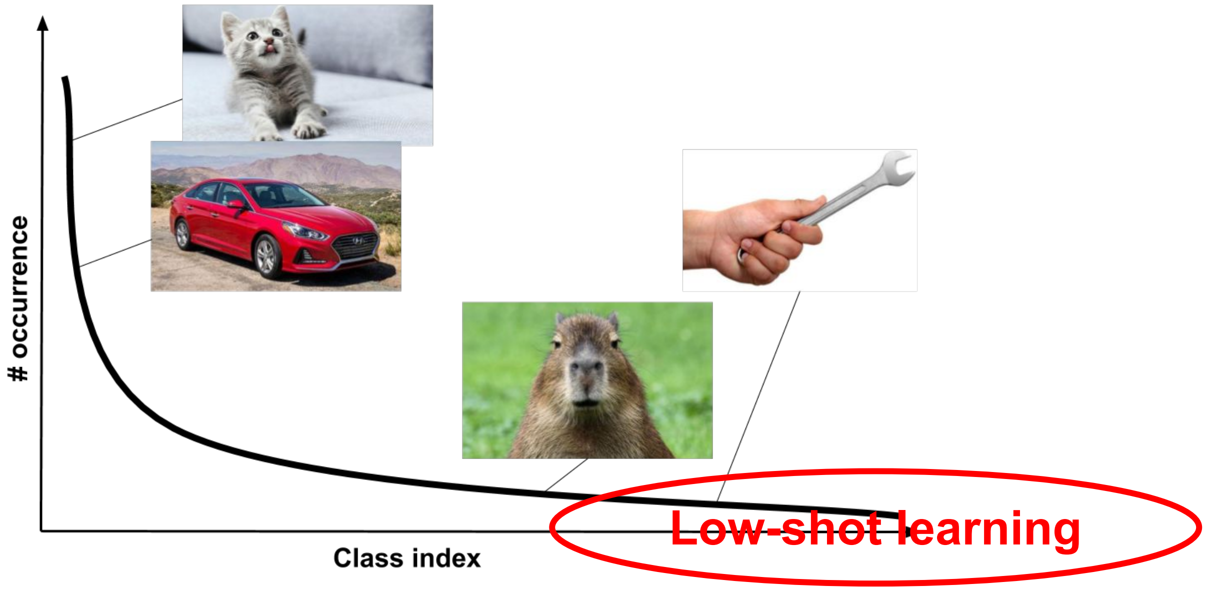
- Introduction & motivation
- Zero-shot learning
  - Definition
  - Side information
  - Zero-shot learning models
  - Exercise
- Low-shot learning
  - Definition
  - Low-shot learning models
- Tips & tricks
- Exercises

Low-shot Learning

Structure

- Introduction & motivation
- Zero-shot learning
  - Definition
  - Side information
  - Zero-shot learning models
  - Exercise
- Low-shot learning
  - Definition
  - Low-shot learning models
- Tips & tricks
- Exercises

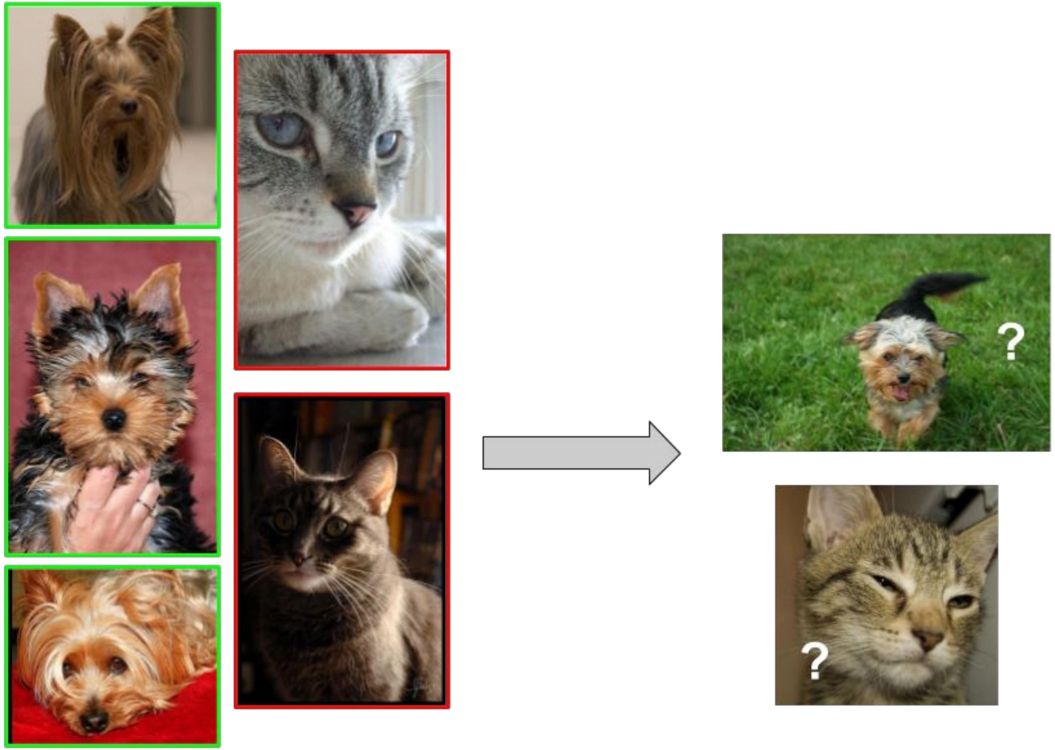
Go back to tail distribution..



https://www.cars.com/
http://www.foxnews.com/lifestyle/2017/11/09/how-to-keep-cat-from-scratching-your-sofa-to-shreds.html
https://www.livescience.com/55223-capybara-facts.html
https://www.indiamart.com/proddetail/hand-wrench-13045857897.html

Low-shot learning

- Ability to generalize only with a few examples
- Exploits prior learning on other classes



http://www.image-net.org/

Structure

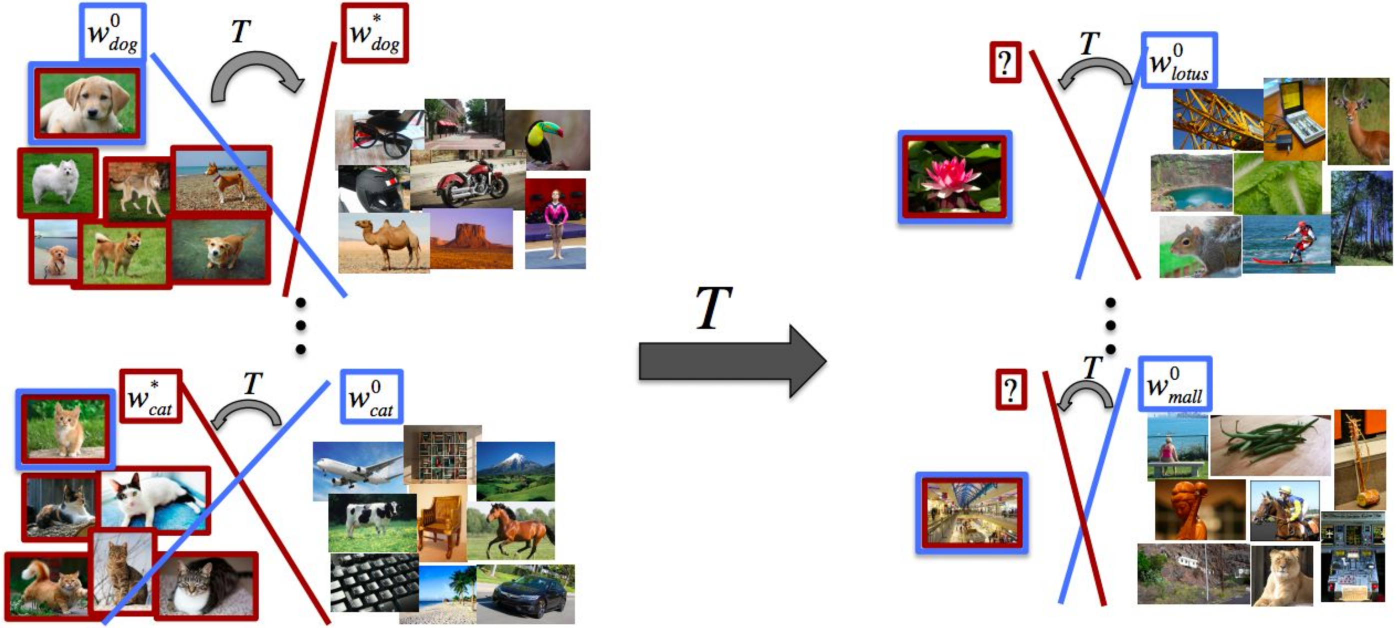
- Introduction & motivation
- Zero-shot learning
  - Definition
  - Side information
  - Zero-shot learning models
  - Exercise
- Low-shot learning
  - Definition
  - Low-shot learning models
- Tips & tricks
- Exercises

Low-shot learning approaches

We will overview three recent works on the problem of low-shot image classification

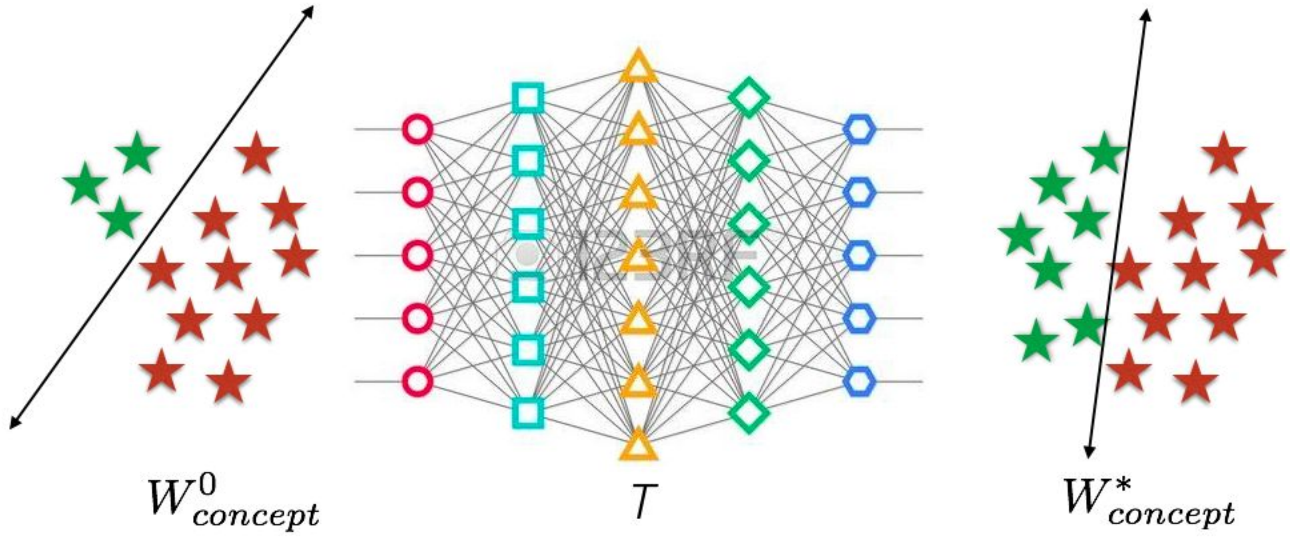
- Learning to learn	Carnegie Mellon University
- Matching nets	Google Deepmind
- Shrinking and Hallucinating Features	Facebook AI

Learning to learn



[Wang and Hebert ECCV2016]

Learning to learn



[Wang and Hebert ECCV2016]

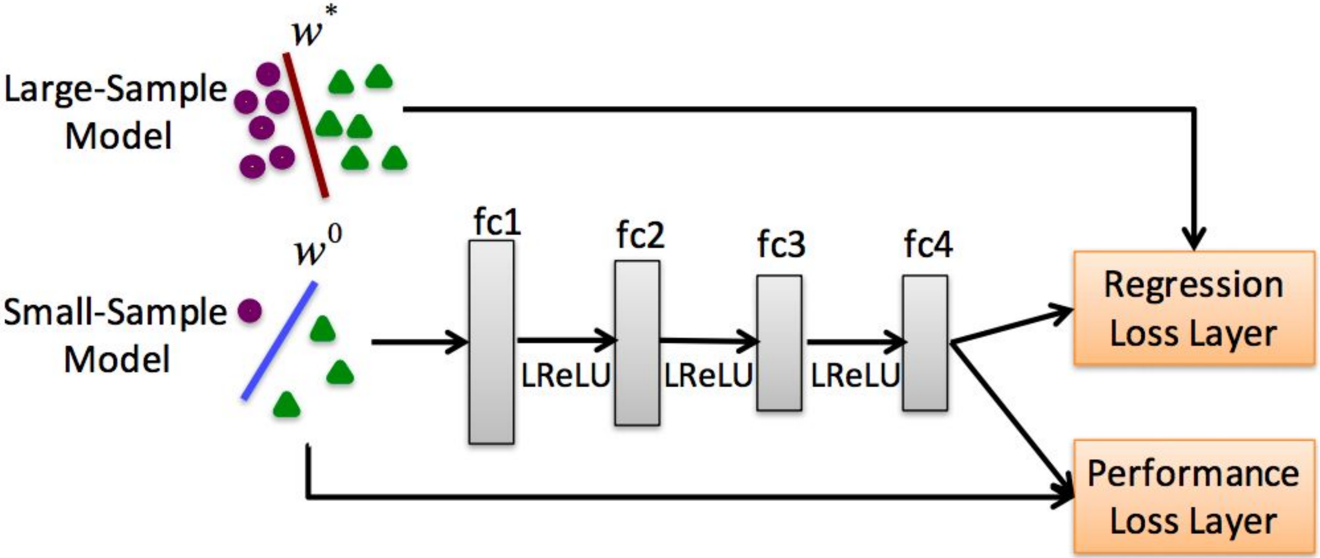
Learning to learn

Loss function

$$L(\Theta) = \sum_{j=1}^{J} \left\{ \underbrace{\frac{1}{2} \left\| \mathbf{w}_j^* - T\left(\mathbf{w}_j^0, \Theta\right) \right\|_2^2}_{\substack{\text{Model regression term} \\ \text{Euclidean distance}}} + \lambda \sum_{i=1}^{M+N} \underbrace{\left[1 - y_i^j \left(T\left(\mathbf{w}_j^0, \Theta\right)^T \mathbf{x}_i^j \right) \right]_+}_{\substack{\text{Data fitting term} \\ \text{Hinge loss}}} \right\}$$

[Wang and Hebert ECCV2016]

Learning to learn



[Wang and Hebert ECCV2016]

Learning to learn

Novel categories:

- **Initialization**
  learn model from small set of K (image,label) pairs
- **Transformation**
  perform the learned transformation T
- **Refinement**
  retrain SVM using the transformed model as regularizer

$$R(\mathbf{w}) = \frac{1}{2}\left\|\mathbf{w} - T\left(\mathbf{w}^0, \Theta\right)\right\|_2^2 + \eta \sum_{i=1}^{K} \left[1 - y_i\left(\mathbf{w}^T \mathbf{x}_i\right)\right]_+$$

[Wang and Hebert ECCV2016]

Matching Networks

Given a support set $S = \{(x_i, y_i)\}_{i=1}^{k}$ learns the mapping $S \rightarrow C_S(x)$

The classifier defines the probability distribution, P is parameterized by a neural network:

$$C(x^{test}) = P(y^{test}|x^{test}, S)$$

Prediction:

$$argmax_y P(y|x^{test}, S)$$

[Vinyals, et al. NIPS2016]

Matching Networks

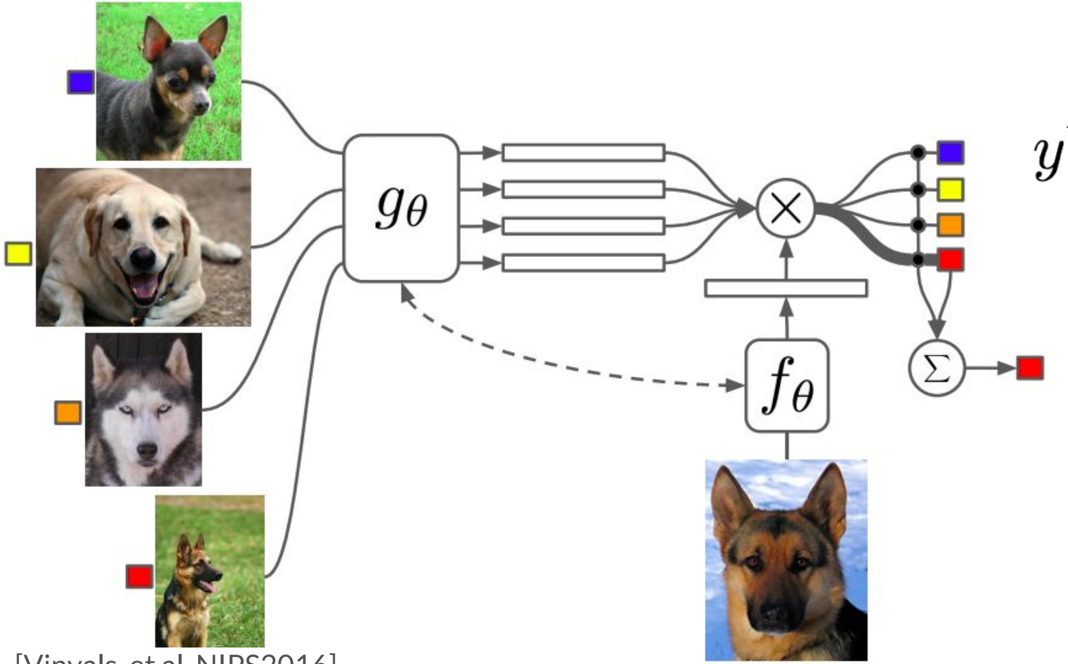
Prediction

$$y^{test} = \sum_{i=1}^{k} a(x^{test}, x_i) y_i$$

- Attention mechanism
- Linear combination of support set labels

[Vinyals, et al. NIPS2016]

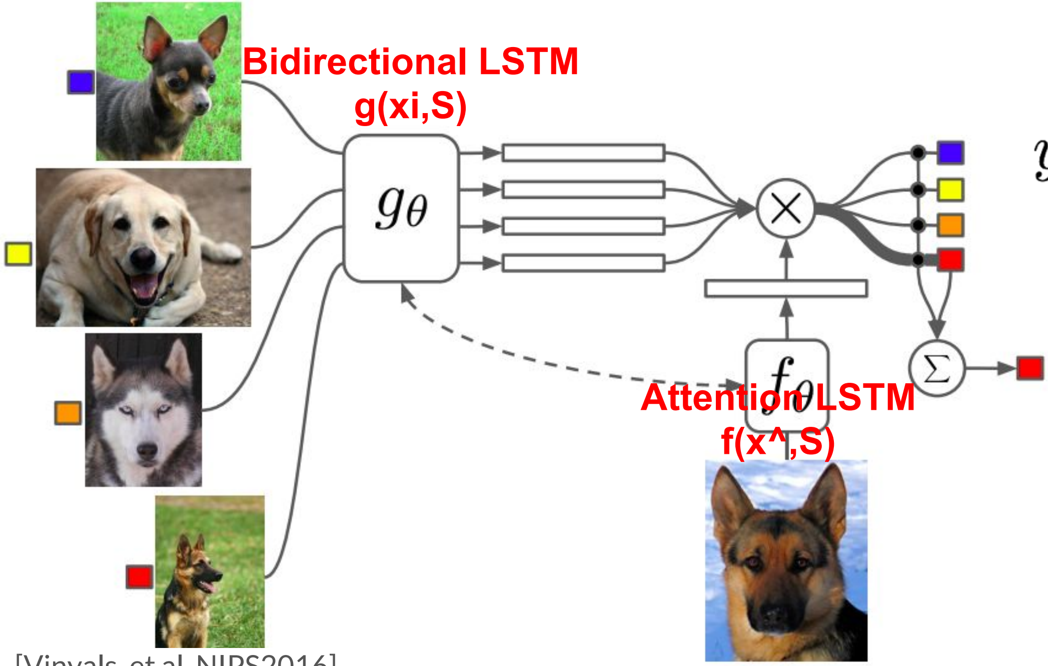
Matching Networks



$$y^{test} = \sum_{i=1}^{k} a(x^{test}, x_i)y_i$$

[Vinyals, et al. NIPS2016]

Matching Networks



$$y^{test} = \sum_{i=1}^{k} a(x^{test}, x_i) y_i$$

0.1*Chihuahua
0.1*Labrador Retriever
0.5*German Shepherd
0.3*Siberian Husky

[Vinyals, et al. NIPS2016]

Matching Networks

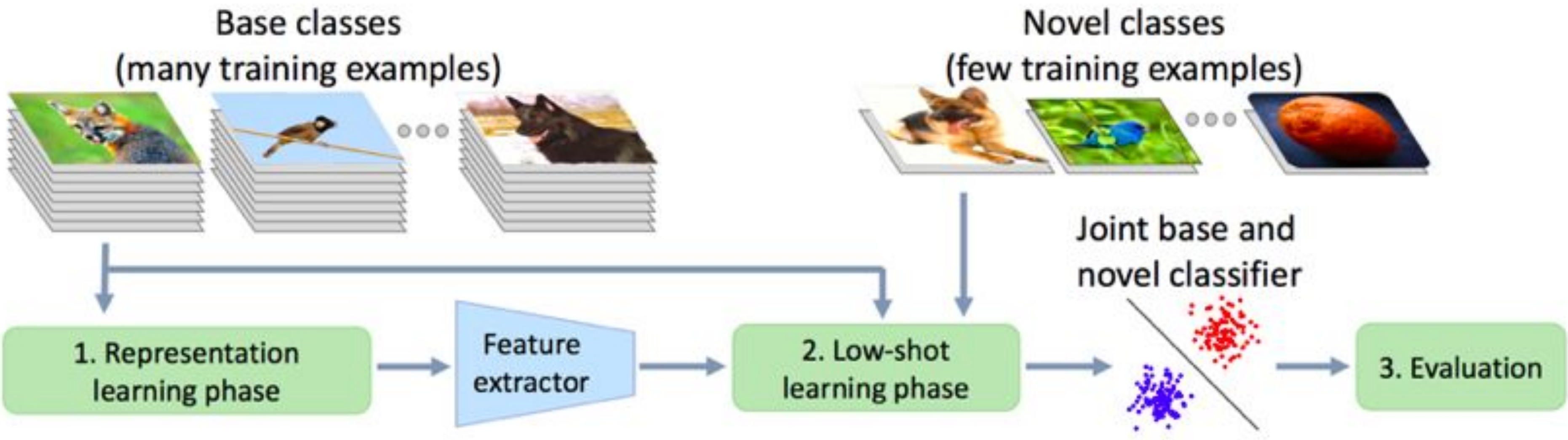
Training strategy

1. Sample task **T** (5 labels, up to 5 examples per label)
2. Sample a label set **L** from T e.g. {cats, dogs}
3. Sample a support set **S** examples from L
4. Sample batch **B** examples from L
5. Evaluate loss on **B** using **S**

$$\theta = \underset{\theta}{argmax} E_{L \sim T}[E_{S \sim L, B \sim L}[\sum_{(x,y) \in B} log P_\theta(y|x, S)]]$$

[Vinyals, et al. NIPS2016]

Shrinking and Hallucinating Features



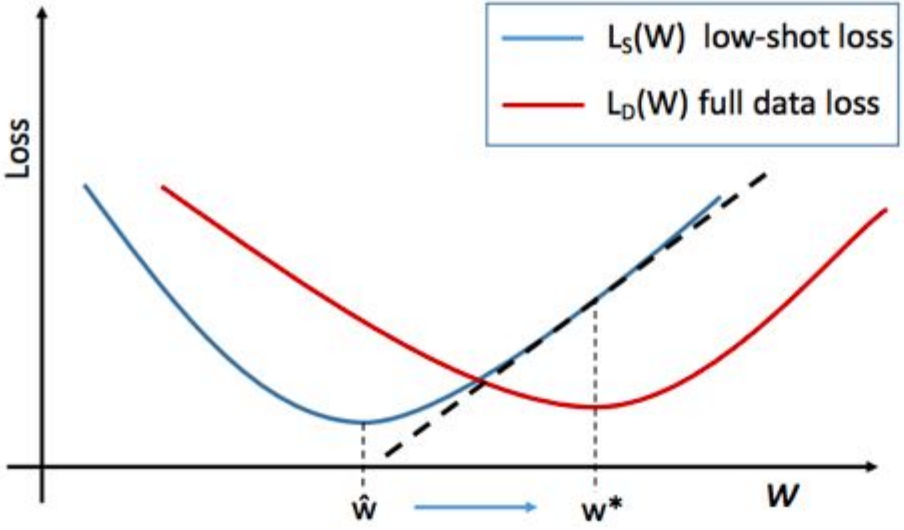
[Hariharan and Girshick arXiv2016]

Shrinking and Hallucinating Features

Introduces Squared Gradient Magnitude loss

- Minimise the loss of low-shot during representation learning

→ better representation for low-shot learning



[Hariharan and Girshick arXiv2016]

Shrinking and Hallucinating Features

Train feature extractor and classifier on D (all data) has the objective

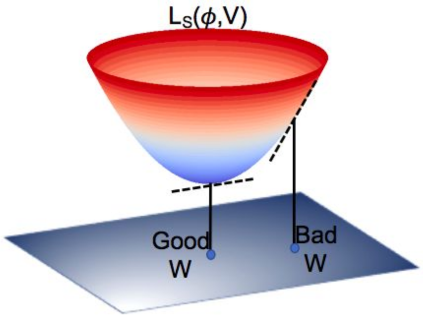
$$\min_{W,\phi} L_D(\phi, W) = \min_{W,\phi} \frac{1}{|D|} \sum_{(x,y)\in D} L_{cls}(W, \phi(x), y)$$

For small set S, the objective

$$\min_{V} L_S(\phi, V) = \min_{V} \frac{1}{|S|} \sum_{(x,y)\in S} L_{cls}(V, \phi(x), y)$$

Minimise

$$\tilde{L}_S(\phi, W) = \|\nabla_V L_S(\phi, V)|_{V=W}\|^2$$



[Hariharan and Girshick arXiv2016]

Shrinking and Hallucinating Features

$$\tilde{L}_S(\phi, W) = \sum_{k=1}^{K}(p_k(W, \phi(x)) - \delta_{yk})^2 \|\phi(x)\|^2$$
$$= \alpha(W, \phi(x), y)\|\phi(x)\|^2.$$

$\alpha(W, \phi(x), y)$ Per example weight that is higher for misclassified data points

Final SGM loss

$$L_D^{SGM}(\phi, W) = \frac{1}{|D|} \sum_{(x,y)\in D} \alpha(W, \phi(x), y)\|\phi(x)\|^2$$

[Hariharan and Girshick arXiv2016]

Shrinking and Hallucinating Features

Train feature representation by minimizing a linear combination of the SGM loss and the original classification objective

$$\min_{W,\phi} L_D(\phi, W) + \lambda L_D^{SGM}(\phi, W)$$

[Hariharan and Girshick arXiv2016]

Shrinking and Hallucinating Features

Hallucinate samples

perched bird with sky background

perched bird with green background

Assumption

Any two examples z1 and z2 belonging to the same category represent a plausible transformation.

→ Given a novel category example x, apply to x the transformation that sent z1 to z2.

[Hariharan and Girshick arXiv2016]

Shrinking and Hallucinating Features

Fully supervised regression using MLP of 3 fully connected layers

[Hariharan and Girshick arXiv2016]

Shrinking and Hallucinating Features

Fully supervised regression using MLP of 3 fully connected layers

1. Cluster feature vectors in each category into 100 clusters.

[Hariharan and Girshick arXiv2016]

Shrinking and Hallucinating Features

Fully supervised regression using MLP of 3 fully connected layers

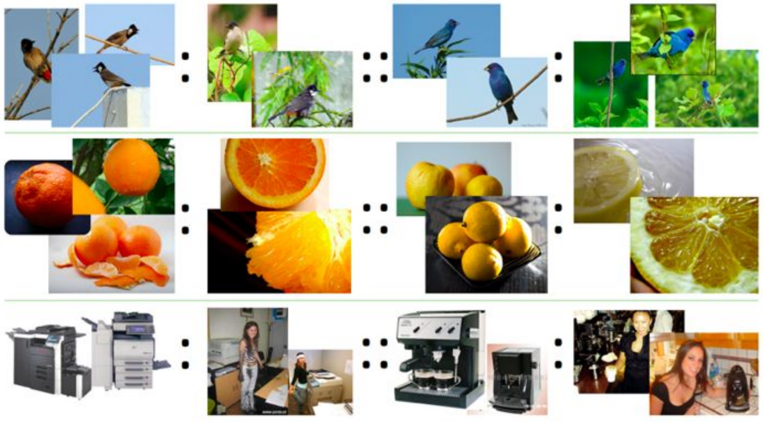
1. Cluster feature vectors in each category into 100 clusters.
2. Form quadruple of centroids (2 centroids from 2 classes)

[Hariharan and Girshick arXiv2016]

Shrinking and Hallucinating Features

Fully supervised regression using MLP of 3 fully connected layers

1. Cluster feature vectors in each category into 100 clusters.
2. Form quadruple of centroids (2 centroids from 2 classes).
3. Feed 3 centroids and predict the forth.

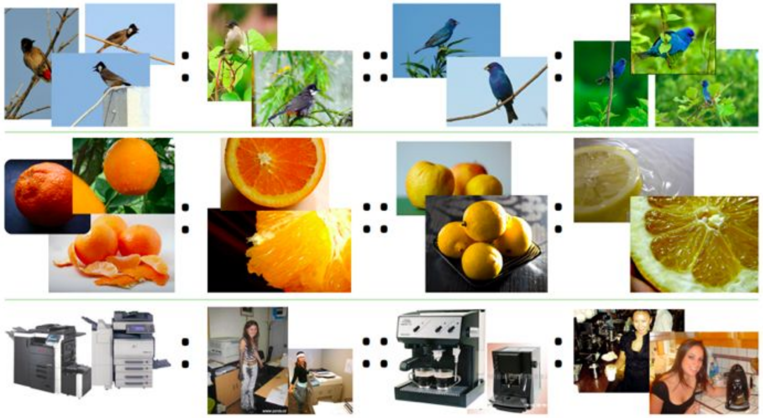


[Hariharan and Girshick arXiv2016]

Shrinking and Hallucinating Features

Fully supervised regression using MLP of 3 fully connected layers

1. Cluster feature vectors in each category into 100 clusters.
2. Form quadruple of centroids (2 centroids from 2 classes).
3. Feed 3 centroids and predict the forth.
4. Minimize the weighted sum of two losses:

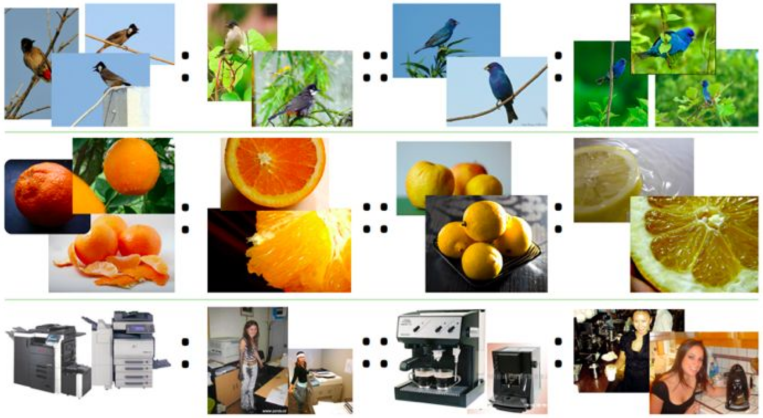


[Hariharan and Girshick arXiv2016]

Shrinking and Hallucinating Features

Fully supervised regression using MLP of 3 fully connected layers

1. Cluster feature vectors in each category into 100 clusters.
2. Form quadruple of centroids (2 centroids from 2 classes).
3. Feed 3 centroids and predict the forth.
4. Minimize the weighted sum of two losses:
    - Classification loss
    - Mean squared error



[Hariharan and Girshick arXiv2016]

Results

Representation	Lowshot phase	n=1	2	5	10	20
ResNet-10						
Baseline	Classifier	14.1	33.3	56.2	66.2	71.5
Baseline	Generation* + Classifier	29.7	42.2	56.1	64.5	70.0
SGM*	Classifier	23.1	42.4	*61.7*	69.6	*73.8*
SGM*	Generation* + Classifier	*32.8*	46.4	*61.7*	*69.7*	*73.8*
L2*	Classifier	29.1	*47.4*	**62.3**	68.0	70.6
Baseline	Model Regression [47]	20.7	39.4	59.6	68.5	73.5
Baseline	Matching Network [46]	**41.3**	**51.3**	*62.1*	67.8	71.8

Top-5 accuracy on Imagenet1K for novel classes only

[Hariharan and Girshick arXiv2016]

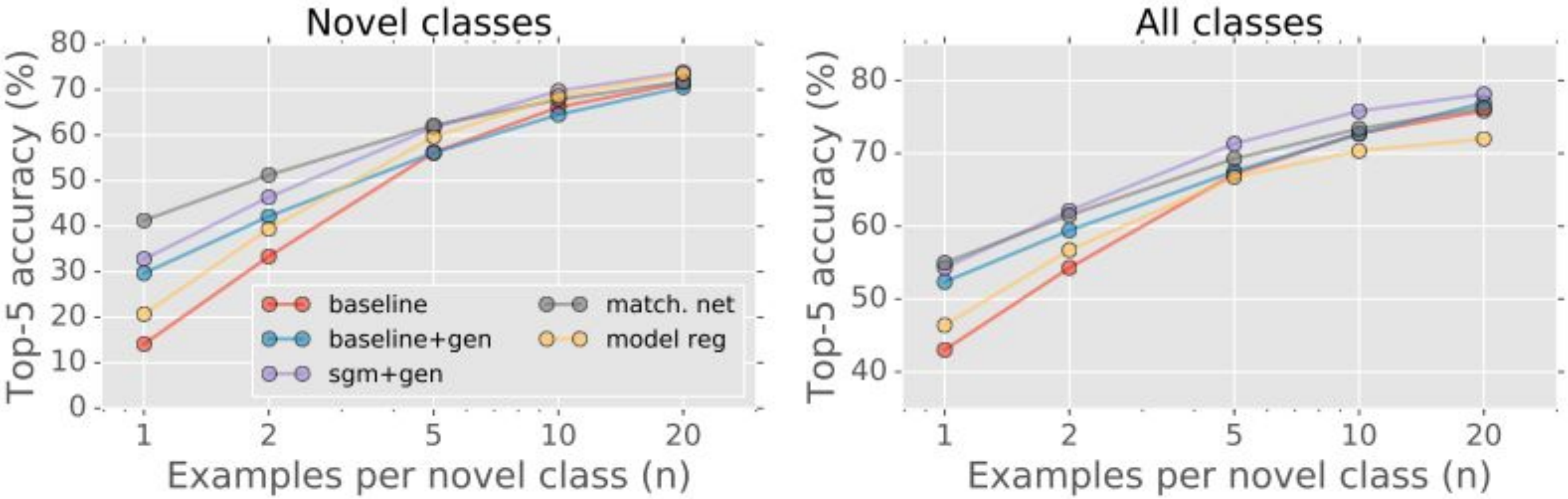
Results

Representation	Lowshot phase	n=1	2	5	10	20
ResNet-10						
Baseline	Classifier	43.0	54.3	67.2	72.8	75.9
Baseline	Generation* + Classifier	52.4	59.4	67.5	72.6	76.9
SGM*	Classifier	49.4	60.5	*71.3*	**75.8**	*78.1*
SGM*	Generation* + Classifier	*54.3*	*62.1*	*71.3*	**75.8**	*78.1*
L2*	Classifier	52.7	**63.0**	**71.5**	74.8	76.4
Baseline	Model Regression [47]	46.4	56.7	66.8	70.4	72.0
Baseline	Matching Network [46]	**55.0**	61.5	69.3	73.4	76.2

Top-5 accuracy on Imagenet1K for all classes

[Hariharan and Girshick arXiv2016]

Results



[Hariharan and Girshick arXiv2016]

Q & A