1711.06025 - Learning to Compare: Relation Network for Few-Shot Learning

- **Yunqiu Xu**
- Related reading:
    - 1706.09529 Learning to Learn: Meta-Critic Networks for Sample Efficient Learning
    - 1710.03463 - Learning to Generalize: Meta-Learning for Domain Generalization
- Brief introduction:
    - Meta learning for few-shot classification
    - What to meta learn: embedding, as well as distance metric to measure similarity

---

1. Introduction

- Challenges for few-shot learning:
    - Complex inference mechanisms
    - Complex RNN
    - Fine tuning the target problem :
        - 这里例子是 MAML 和 Larochelle 的 LSTM-based meta learner, 注意对比下
        - 说他们的缺点是fine-tuning可能不够快?
- Our method:
    - Similar to train an metric: 区别是他们着重于训练transferrable embedding, 但metric是固定的(e.g. 欧式距离), 而我们把metric也作为学习目标之一
    - Two-branch Relation Network:
        - Embedding module: generate representations (embedding) of the query and training images
        - Relation module: compare embedding pairs to check whether thay are from matching classes or not
        - 之前的工作固定了metric, 而且是linear comparator, 这里我们学习非线性的

2. Related Work

- **Learning to Fine-Tune, 这里作者举了两个例子**:
  - MAML, LSTM-based learner: 这里说这份工作强于MAML, 因为不仅仅训练了初始化参数, 还训练了优化器 → 这样效率更低吧?
  - 这两个工作的缺点: fine-tune on the target problem
  - 我们的工作不需要model updates, 就直接feed forward就行
- **RNN Memory based, 我之前看过ICML2017上的meta network**:
  - RNN: knowledge is accumulated in hidden activations/external memory to solve the problem
  - Drawback: 需要保证memory存储了所有或至少是long term的历史信息而没有遗忘, 一方面不容易得到所有信息, 另一方面会占用很大空间
  - 我们的工作使用feed forward CNN, 避免了RNN
- **Embedding and Metric Learning**
  - Embedding: parameterise the weights of feed-forward classifier, 这里meta学得是参数化网络, 给定样本集合, 试图参数化一个分类器
  - Metric learning: learn distance evaluation metric, 这里meta学得是量度
  - 相关工作: prototypical network, siamese network, 重点在学习embedding, 而分类则直接用KNN或者线性分类器量度相似性
  - 我们的工作不固定metric, 而是把metric也当成一个学习目标, 且不限制为线性, 试图学习非线性的分类器
    - 和siamese network相比, 为episodic training strategy
    - 和prototypical network相比, 避免了复杂的RNN embedding
- **Zero-Shot Learning: 这里先略过**

3. Methodology

3.1 Problem definition

- Goal: few-shot classification, recall C-way K-shot is C classes, and K samples for each class
- For each training iteration, sample C classes from train set with K samples for each

class → Sample set $S = \{(x_i, y_i)\}_{i=1}^{K \times C}$

3.2 Model

- **这里先只考虑one-shot, zero-shot 先略过**
- Relation Network:
  - Embedding module f_ψ : input images, output feature map
  - Combine (concat) feature map of sample pair
  - Relation module g_ϕ: input concated feature map, output similarity between these two samples
- C-way 1-shot: 分辨当前样本 x_j 与每一类的样本 x_i 的相似度 $r_{i,j}$

$$r_{i,j} = g_\phi(C(f_\psi(x_i), f_\psi(x_j))), i = 1, 2, \ldots, C$$

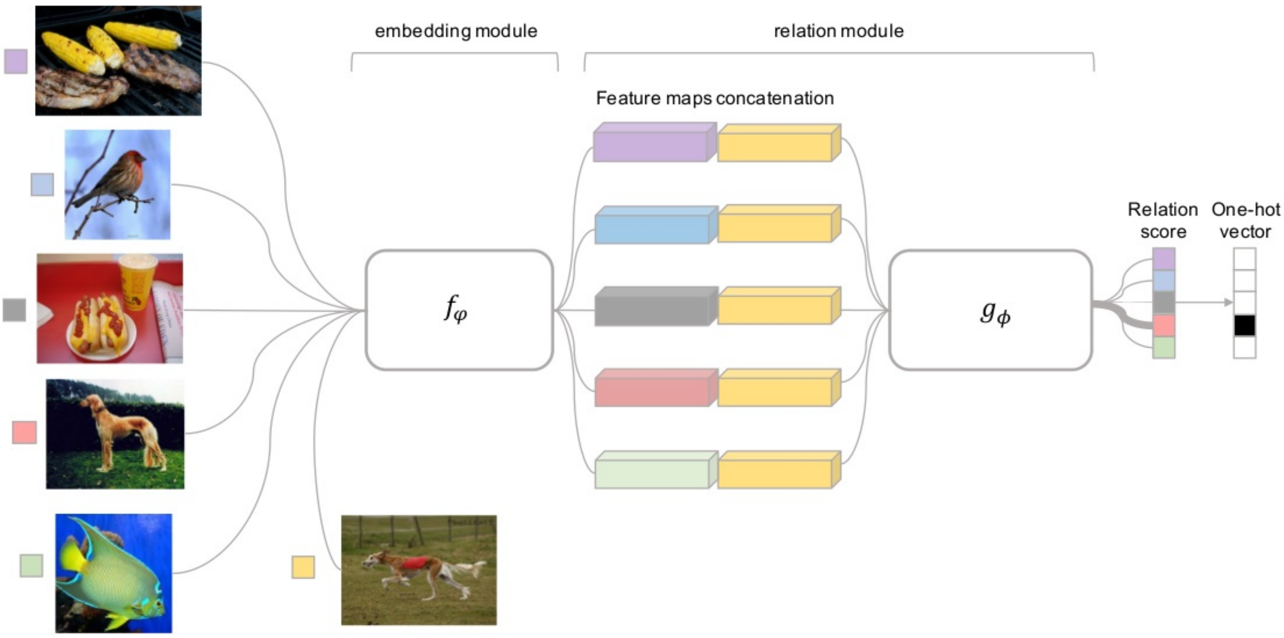

Figure 1: Relation Network architecture with a 5-way 1-shot 1-query example.

3.3 Network Architecture

- Naive Network有点像VGG, Deeper有点像ResNet

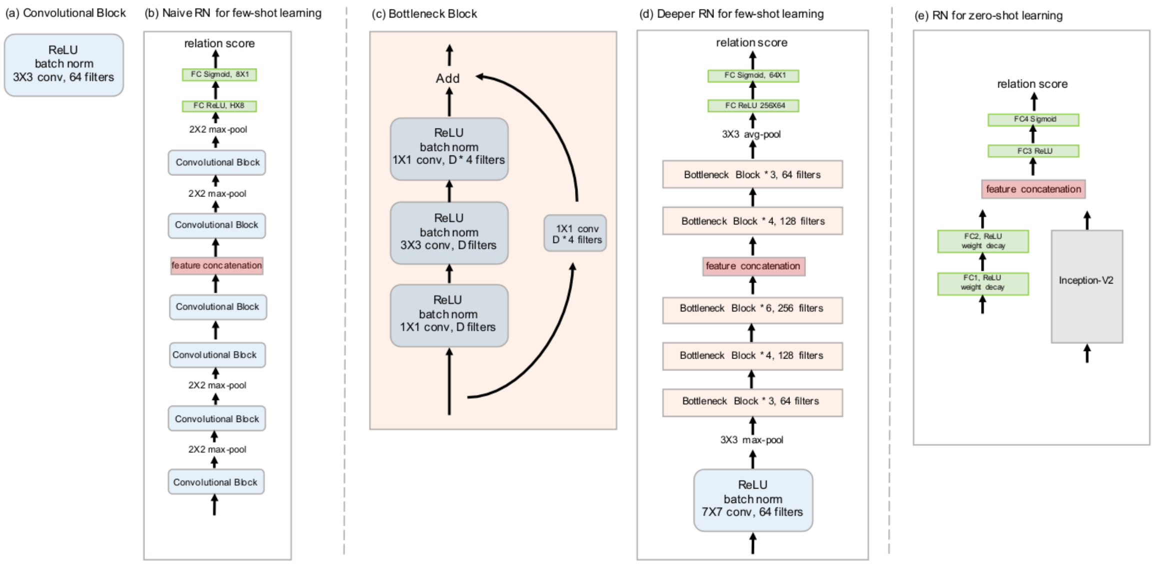

Figure 2: Relation Network architecture for few-shot learning: (b) naive version, (d) deeper version. Relation Network architecture for (e) zero-shot learning. These are composed of elements including (a) convolutional block, and (b) bottleneck block.

4. Experiment

- 还是经典的Omniglot和MiniImageNet

Model	Fine Tune	5-way Acc.		20-way Acc.	
		1-shot	5-shot	1-shot	5-shot
MANN [31]	N	82.8%	94.9%	-	-
CONVOLUTIONAL SIAMESE NETS [18]	N	96.7%	98.4%	88.0%	96.5%
CONVOLUTIONAL SIAMESE NETS [18]	Y	97.3%	98.4%	88.1%	97.0%
MATCHING NETS [38]	N	98.1%	98.9%	93.8%	98.5%
MATCHING NETS [38]	Y	97.9%	98.7%	93.5%	98.7%
SIAMESE NETS WITH MEMORY [16]	N	98.4%	99.6%	95.0%	98.6%
NEURAL STATISTICIAN [8]	N	98.1%	99.5%	93.2%	98.1%
META NETS [26]	N	99.0%	-	97.0%	-
PROTOTYPICAL NETS [35]	N	98.8%	99.7%	96.0%	98.9%
MAML [10]	Y	98.7 $\pm$ 0.4%	**99.9 $\pm$ 0.1%**	95.8 $\pm$ 0.3%	98.9 $\pm$ 0.2%
RELATION NET	N	**99.6 $\pm$ 0.2%**	**99.8$\pm$ 0.1%**	**97.6 $\pm$ 0.2%**	**99.1$\pm$ 0.1%**

Table 1: Omniglot few-shot classification. Results are accuracies averaged over 1000 test episodes and with 95% confidence intervals where reported. The best-performing method is highlighted, along with others whose confidence intervals overlap. ‘-’: not reported.

Model	FT	5-way Acc.	
		1-shot	5-shot
MATCHING NETS [38]	N	43.56 ± 0.84%	55.31 ± 0.73%
META NETS [26]	N	49.21 ± 0.96%	-
META-LEARN LSTM [28]	N	43.44 ± 0.77%	60.60 ± 0.71%
MAML [10]	Y	48.70 ± 1.84%	63.11 ± 0.92%
PROTOTYPICAL NETS [35]	N	49.42 ± 0.78%	**68.20 ± 0.66%**
RELATION NET (NAIVE)	N	**51.38 ± 0.82%**	**67.07 ± 0.69%**
TCML [25]	N	55.71 ± 0.99%	68.88 ± 0.92%
RELATION NET (DEEPER)	N	**57.02 ± 0.92%**	**71.07 ± 0.69%**

Table 2: Few-shot classification accuracies on *mini*Imagenet. All accuracy results are averaged over 600 test episodes and are reported with 95% confidence intervals, same as [35]. For each task, the best-performing method is highlighted, along with any others whose confidence intervals overlap. '-': not reported.

5. Why does it work?

- 前人的工作:
  - pre-specified metric(e.g. 基于欧式距离), 学习feature embedding, 而metric是固定的
  - Conventional metric learning: 学习简单的曼哈顿metric, 固定feature representation
- 我们的工作: 同时学习embedding, non-linear metric (相似度方程)
- 为什么我们的工作重要:
  - 我们自己学习选择合适的metric而不是手动指定
  - 前人工作固定metric, 需要假定特征可以被element-wise比较, 且假定在embedding后是线性可分的. 因此就非常依赖学到的embedding network, 如果这个网络生成的embedding不足以表达特征, 就傻逼了

    - 而我们的工作同时学习非线性的相似度度量以及embedding, 可以更好地区分匹配/不匹配样本对

6. Summary

- 对比前人的工作, 本工作把metric也作为学习的目标, 相当于让模型根据任务选择相似度度量咯
- 感觉创新性一般, 不过可以看下其思路, 现在网上也有相关实现可以学习下