

Task Agnostic Meta-Learning for Few-Shot Learning

Muhammad Abdullah Jamal
 University of Central Florida
 Orlando, FL
 a_jamal@knights.ucf.edu

Guo-Jun Qi *
 Laboratory for MACHine
 Perception and LEARNING (MAPLE)
<http://maple-lab.net/>
 Huawei Cloud
 guojunqi@gmail.com

Abstract

Meta-learning approaches have been proposed to tackle the few-shot learning problem. Typically, a meta-learner is trained on a variety of tasks in the hopes of being generalizable to new tasks. However, the generalizability on new tasks of a meta-learner could be fragile when it is over-trained on existing tasks during meta-training phase. In other words, the initial model of a meta-learner could be too biased towards existing tasks to adapt to new tasks, especially when only very few examples are available to update the model. To avoid a biased meta-learner and improve its generalizability, we propose a novel paradigm of Task-Agnostic Meta-Learning (TAML) algorithms. Specifically, we present an entropy-based approach that meta-learns an unbiased initial model with the largest uncertainty over the output labels by preventing it from over-performing in classification tasks. Alternatively, a more general inequality-minimization TAML is presented for more ubiquitous scenarios by directly minimizing the inequality of initial losses beyond the classification tasks wherever a suitable loss can be defined. Experiments on benchmarked datasets demonstrate that the proposed approaches outperform compared meta-learning algorithms in both few-shot classification and reinforcement learning tasks.

1. Introduction

The key to achieving human level intelligence is to learn from a few labeled examples. Human can learn and adapt quickly from a few examples using prior experience. We want our learner to be able to learn from a few examples and quickly adapt to a changing task. All these concerns motivate to study the few-shot learning problem. The advantage of studying the few-shot problem is that it only relies on few examples and it alleviates the need to collect large amount

of labeled training set which is a cumbersome process.

Recently, meta-learning approach is being used to tackle the problem of few-shot learning. A meta-learning model usually contains two parts – an initial model, and an updating strategy (e.g., a parameterized model) to train the initial model to a new task with few examples. Then the goal of meta-learning is to automatically meta-learn the optimal parameters for both the initial model and the updating strategy that are generalizable across a variety of tasks. There are many meta-learning approaches that show promising results on few-shot learning problems. For example, Meta-LSTM [1] uses LSTM meta-learner that not only learns initial model but also the updating rule. On the contrary, MAML [2] only learns an initial model since its updating rule is fixed to a classic gradient descent method as a meta-learner.

The problem with existing meta-learning approaches is that the initial model can be trained biased towards some tasks, particularly those sampled in meta-training phase. Such a biased initial model may not be well generalizable to an unseen task that has a large deviation from meta-training tasks, especially when very few examples are available on the new task. This inspires us to meta-train an unbiased initial model by preventing it from overperforming on some tasks or directly minimizing the inequality of performances across different tasks, in a hope to make it more generalizable to unseen tasks. To this end, we propose a Task-Agnostic Meta-Learning (TAML) algorithms in this paper.

Specifically, we propose two novel paradigms of TAML algorithms – an entropy-based TAML and inequality-minimization measures based TAML. The idea of using entropy based approach is to maximize the entropy of labels predicted by the initial model to prevent it from overperforming on some tasks. However, the entropy-based approach is limited to discrete outputs from a model, making it more amenable to classification tasks.

The second paradigm is inspired by inequality measures used in Economics. The idea is to meta-train an initial model in such a way that it directly minimizes the inequality of

*Corresponding author: G.-J. Qi.

losses by the initial model across a variety of tasks. This will force the meta-learner to learn a unbiased initial model without over-performing on some particular tasks. Meanwhile, any form of losses can be adopted for involved task without having to rely on discrete outputs. This makes this paradigm more ubiquitous to many scenarios beyond classification tasks.

The remainder of the paper is organized as follows. We review about the related work in Section 2. It is followed by the elaboration of the proposed TAML approach to meta-learning in Section 3. In Section 4, we present extensive experimental studies on few-shot classification and reinforcement learning.

2. Related Work

The idea of meta-learning has been proposed more than a couple of decades ago [3, 4, 5]. Most of the approaches to meta-learning include learning a learner’s model by training a meta-learner. Recent studies towards meta-learning for deep neural networks include learning a hand-designed optimizer like SGD by parameterizing it through recurrent neural networks. Li [6], and Andrychowicz [7] studied a LSTM based meta-learner that takes the gradients from learner and performs an optimization step. Recently, meta-learning framework has been used to solve few-shot classification problems. [1] used the same LSTM based meta-learner approach in which LSTM meta-learner takes the gradient of a learner and proposed an update to the learner’s parameters. The approach learns both weight initialization and an optimizer of the model weights. Finn [2] proposed a more general approach for meta-learning known as MAML by simply learning weight initialization for a learner through a fixed gradient descent. It trains a model on a variety of tasks to have a good initialization point that can be quickly adapted (few or one gradient steps) to a new task using few training examples. Meta-SGD [8] extends the MAML, which not only learns weight initialization but also the learner’s update step size. [9] proposes a temporal convolution and attention based meta-learner called SNAIL that achieves state-of-the-art performance for few-shot classification tasks and reinforcement learning tasks.

Other paradigms of meta-learning approaches include training a memory augmented neural network on existing tasks by coupling with LSTM or feed-forward neural network controller [10, 11]. There are also several non-meta-learning approaches to few-shot classification problem by designing specific neural architectures. For example, [12] trains a Siamese network to compare new examples with existing ones in a learned metric space. Vinyals [13] used a differentiable nearest neighbour loss by utilizing the cosine similarities between the features produced by a convolutional neural network. [14] proposed a similar approach to matching net but used a square euclidean distance metric

instead. In this paper, we mainly focus on the meta-learning approaches and their applications to few-shot classification and reinforcement tasks.

3. Approach

Our goal is to train a model that can be task-agnostic in a way that it prevents the initial model or learner to over-perform on a particular task. In this section, we will first describe our entropy based and inequality-minimization measures based approach to the problem, and then we will discuss some of the inequality measures that we used in the paper.

3.1. Task Agnostic Meta-Learning

In this section, we propose a task-agnostic approach for few-shot meta-learning. The goal of few-shot meta-learning is to train a model in such a way that it can learn to adapt rapidly using few samples for a new task. In this meta-learning approach, a learner is trained during a meta-learning phase on variety of sampled tasks so that it can learn new tasks, while a meta-learner trains the learner and is responsible for learning the update rule and initial model.

The problem with the current meta-learning approach is that the initial model or learner can be biased towards some tasks sampled during the meta-training phase, particularly when future tasks in the test phase may have discrepancy from those in the training tasks. In this case, we wish to avoid an initial model over-performing on some tasks. Moreover, an over-performed initial model could also prevent the meta-learner to learn a better update rule with consistent performance across tasks.

To address this problem, we impose an unbiased task-agnostic prior on the initial model by preventing it from over-performing on some tasks so that a meta-learner can achieve a more competitive update rule. There have been many meta-learning approaches to few-shot learning problems that have been briefly discussed in the section 2. While the task-agnostic prior is a widely applicable principle for many meta-learning algorithms, we mainly choose Model-Agnostic Meta Learning approach (MAML) as an example to present the idea, and it is not hard to extend to other meta-learning approaches.

In the following, we will depict the idea by presenting two paradigms of task-agnostic meta-learning (TAML) algorithms – the entropy-maximization/reduction TAML and inequality-minimization TAML.

3.1.1 Entropy-Maximization/Reduction TAML

For simplicity, we express the model as a function f_θ that is parameterized by θ . For example, it can be a classifier that takes an input example and outputs its discrete label. During meta-training, a batch of tasks are sampled from a task

distribution $p(\mathcal{T})$, and each task is K -shot N -way problem where K represents the number of training examples while N represent the number of classes depending on the problem setting. In the MAML, a model is trained on a task \mathcal{T}_i using K examples and then tested on a few new examples \mathcal{D}_{val} for this task.

A model has an initial parameter θ and when it is trained on the task \mathcal{T}_i , its parameter is updated from θ to θ_i by following an updating rule. For example, for K -shot classification, stochastic gradient descent can be used to update model parameter by $\theta_i \leftarrow \theta - \alpha \nabla_{\theta} \mathcal{L}_{\mathcal{T}_i}(f_{\theta})$ that attempts to minimize the cross-entropy loss $\mathcal{L}_{\mathcal{T}_i}(f_{\theta})$ for the classification task \mathcal{T}_i over K examples.

To prevent the initial model f_{θ} from over-performing on a task, we prefer it makes a random guess over predicted labels with an equal probability so that it is not biased towards the task. This can be expressed as a maximum-entropy prior over θ so that the initial model should have a large entropy over the predicted labels over samples from task \mathcal{T}_i .

The entropy for task \mathcal{T}_i is computed by sampling x_i from $P_{\mathcal{T}_i}(x)$ over its output probabilities $y_{i,n}$ over N predicted labels:

$$\mathcal{H}_{\mathcal{T}_i}(f_{\theta}) = -\mathbb{E}_{x_i \sim P_{\mathcal{T}_i}(x)} \sum_{n=1}^N \hat{y}_{i,n} \log(\hat{y}_{i,n}) \quad (1)$$

where $[y_{i,1}, \dots, y_{i,N}] = f_{\theta}(x_i)$ is the predictions by f_{θ} , which are often an output from a softmax layer in a classification task. The above expectation is taken over x_i 's sampled from task \mathcal{T}_i .

Alternatively, one can not only maximize the entropy before the update of initial model's parameter, but also minimize the entropy after the update. So overall, we maximize the entropy reduction for each task \mathcal{T}_i as $\mathcal{H}_{\mathcal{T}_i}(f_{\theta}) - \mathcal{H}_{\mathcal{T}_i}(f_{\theta_i})$. The minimization of $\mathcal{H}_{\mathcal{T}_i}(f_{\theta_i})$ means that the model can become more certain about the labels with a higher confidence after updating the parameter θ to θ_i . This entropy term can be combined with the typical meta-training objective term as a regularizer to find the optimal θ , which is

$$\min_{\theta} \mathbb{E}_{\mathcal{T}_i \sim P(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_i}) + \lambda [-\mathcal{H}_{\mathcal{T}_i}(f_{\theta}) + \mathcal{H}_{\mathcal{T}_i}(f_{\theta_i})]$$

where λ is a positive balancing coefficient, and the first term is the expected loss for the updated model f_{θ_i} . The entropy-reduction algorithm is summarized in 1.

Unfortunately, the entropy-based TAML is subject to a critical limitation – it is only amenable to discrete labels in classification tasks to compute the entropy. In contrast, many other learning problems, such as regression and reinforcement learning problems, it is often trained by minimizing some loss or error functions directly without explicitly accessing a particular form of outputs like discrete labels. To make the TAML widely applicable, we need to define an alternative metric to measure and minimize the bias across tasks.

Algorithm 1 TAML for Few-Shot Classification

Require: $p(\mathcal{T})$: distribution over tasks.

Require: α, β : hyperparameters

Randomly Initialize θ

while not done **do**

 Sample batch of tasks $\mathcal{T}_i \sim p(\mathcal{T})$

for all \mathcal{T}_i **do**

 Sample K samples from \mathcal{T}_i

 Evaluate $\nabla_{\theta} \mathcal{L}_{\mathcal{T}_i}(f_{\theta})$ and $\mathcal{L}_{\mathcal{T}_i}(f_{\theta})$ using K samples.

 Compute adapted parameters using gradient descent.

$\theta_i \leftarrow \theta - \alpha \nabla_{\theta} \mathcal{L}_{\mathcal{T}_i}$

 Sample \mathcal{D}_{val} from \mathcal{T}_i for meta update.

end for

if Entropy-Reduction TAML **then**

 Update $\theta \leftarrow \theta - \beta \nabla_{\theta} \{ \mathbb{E}_{\mathcal{T}_i \sim P(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_i}) + \lambda [-\mathcal{H}_{\mathcal{T}_i}(f_{\theta}) + \mathcal{H}_{\mathcal{T}_i}(f_{\theta_i})] \}$ using \mathcal{D}_{val} , $\mathcal{L}_{\mathcal{T}_i}$, and $\mathcal{H}_{\mathcal{T}_i}$.

else if Inequality Measures Based TAML **then**

 Update $\theta \leftarrow \theta - \beta \nabla_{\theta} [\mathbb{E}_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_i}) + \lambda \mathcal{I}_{\mathcal{E}}(\{ \mathcal{L}_{\mathcal{T}_i}(f_{\theta}) \})]$ using $\mathcal{D}_{val,i}$, $\mathcal{L}_{\mathcal{T}_i}$, and $\mathcal{I}_{\mathcal{E}}$

end if

end while

3.1.2 Inequality-Minimization TAML

We wish to train a task-agnostic model in meta-learning such that its initial performance is unbiased towards any particular task \mathcal{T}_i . Such a task-agnostic meta-learner would do so by minimizing the inequality of its performances over different tasks.

To this end, we propose an approach based on a large family of statistics used to measure the "economic inequalities" to measure the "task bias". The idea is that the loss of an initial model on each task \mathcal{T}_i is viewed as an income for that task. Then for the TAML model, its loss inequality over multiple tasks is minimized to make the meta-learner task-agnostic.

Specifically, the bias of the initial model towards any particular tasks is minimized during meta-training by minimizing the inequality over the losses of sampled tasks in a batch. So, given an unseen task during testing phase, a better generalization performance is expected on the new task by updating from an unbiased initial model with few examples. The key difference between both TAMLs lies that for entropy, we only consider one task at a time by computing the entropy of its output labels. Moreover, entropy depends on a particular form or explanation of output function, e.g., the SoftMax output. On the contrary, the inequality only depends on the loss, thus it is more ubiquitous.

The complete algorithm is explained in 1. Formally, consider a batch of sampled tasks $\{\mathcal{T}_i\}$ and their losses

$\{\mathcal{L}_{\mathcal{T}_i}(f_\theta)\}$ by the initial model f_θ , one can compute the inequality measure by $\mathcal{I}_{\mathcal{E}}(\{\mathcal{L}_{\mathcal{T}_i}(f_\theta)\})$ as discussed later. Then the initial model parameter θ is meta-learned by minimizing the following objective

$$\mathbb{E}_{\mathcal{T}_i \sim p(\mathcal{T})} [\mathcal{L}_{\mathcal{T}_i}(f_{\theta_i})] + \lambda \mathcal{I}_{\mathcal{E}}(\{\mathcal{L}_{\mathcal{T}_i}(f_\theta)\})$$

through gradient descent as shown in Algorithm 1. It is worth noting that the inequality measure is computed over a set of losses from sampled tasks. The first term is the expected loss by the model f_{θ_i} after the update, while the second is the inequality of losses by the initial model f_θ before the update. Both terms are a function of the initial model parameter θ since θ_i is updated from θ . In the following, we will elaborate on some choices on inequality measures $\mathcal{I}_{\mathcal{E}}$.

3.2. Inequality Measures

Inequality measures are instrumental towards calculating the economic inequalities in the outcomes that can be wealth, incomes, or health related metrics. In meta-learning context, we use $\ell_i = \mathcal{L}_{\mathcal{T}_i}(f_\theta)$ to represent the loss of a task \mathcal{T}_i , $\bar{\ell}$ represents the mean of the losses over sampled tasks, and M is the number of tasks in a single batch. The inequality measures used in TAML are briefly described below.

Thiel Index [15]. This inequality measure has been derived from redundancy in information theory, which is defined as the difference between the maximum entropy of the data and an observed entropy. Suppose that we have M losses $\{\ell_i | i = 1, \dots, M\}$, then Thiel Index is defined as

$$T_T = \frac{1}{M} \sum_{i=1}^M \frac{\ell_i}{\bar{\ell}} \ln \frac{\ell_i}{\bar{\ell}} \quad (2)$$

Generalized Entropy Index [16]. The relation between information theory and information distribution analysis has been exploited to derive a number of measures for inequality. Generalized Entropy index has been proposed to measure the income inequality. It is not a single inequality measure, but it is a family that includes many inequality measures like Thiel Index, Thiel L etc. For some real value α , it is defined as:

$$GE(\alpha) = \begin{cases} \frac{1}{M\alpha(\alpha-1)} \sum_{i=1}^M \left[\left(\frac{\ell_i}{\bar{\ell}} \right)^\alpha - 1 \right], & \alpha \neq 0, 1, \\ \frac{1}{M} \sum_{i=1}^M \frac{\ell_i}{\bar{\ell}} \ln \frac{\ell_i}{\bar{\ell}}, & \alpha = 1, \\ -\frac{1}{M} \sum_{i=1}^M \ln \frac{\ell_i}{\bar{\ell}}, & \alpha = 0, \end{cases} \quad (3)$$

From the equation, we can see that it does represent a family of inequality measures. When α is zero, it is called a mean log deviation of Thiel L, and when α is one, it is actually Thiel Index. A larger GE α value makes this index more sensitive to differences at the upper part of the distribution, and a smaller α value makes it more sensitive to differences at the bottom of the distribution.

Atkinson Index [17] is another measure for income inequality which is useful in determining which end of the distribution contributed the most to the observed inequality. It is defined as :

$$A_\epsilon = \begin{cases} 1 - \frac{1}{\mu} \left(\frac{1}{M} \sum_{i=1}^M \ell_i^{1-\epsilon} \right)^{\frac{1}{1-\epsilon}}, & \text{for } 0 \leq \epsilon \neq 1, \\ 1 - \frac{1}{\bar{\ell}} \left(\frac{1}{M} \prod_{i=1}^M \ell_i \right)^{\frac{1}{M}}, & \text{for } \epsilon = 1, \end{cases} \quad (4)$$

where ϵ is called "inequality aversion parameter". When $\epsilon = 0$ the index becomes more sensitive to the changes in upper end of the distribution, and when it approaches to 1, the index becomes more sensitive to the changes in lower end of the distribution.

Gini-Coefficient [18] is usually defined as the half of the relative absolute mean difference. In terms of meta-learning, if there are M tasks in a single batch and a task \mathcal{T}_i loss is represented by ℓ_i , then Gini-Coefficient is defined as:

$$G = \frac{\sum_{i=1}^M \sum_{j=1}^M |\ell_i - \ell_j|}{2n \sum_{i=1}^M \ell_i} \quad (5)$$

Gini-coefficient is more sensitive to deviation around the middle of the distribution than at the upper or lower part of the distribution.

Variance of Logarithms [19] is another common inequality measure defined as:

$$V_L(\ell) = \frac{1}{M} \sum_{i=1}^M [\ln \ell_i - \ln g(\ell)]^2 \quad (6)$$

where $g(\ell)$ is the geometric mean of ℓ which is defined as $(\prod_{i=1}^M \ell_i)^{1/M}$. The geometric mean put greater emphasis on the lower losses of the distribution.

4. Experiments

We report experiment results in this section to evaluate the efficacy of the proposed TAML approaches on a variety of few-shot learning problems on classification and reinforcement learning.

Table 1. Few Shot Classification results on Omniglot dataset for fully connected network and convolutional network on 5-way setting, where * means re-run results as there is no general training/test splitting available for Omniglot, thus we re-run compared models with the same splitting used in running the TAML for a fair comparison. The \pm shows 95% confidence interval over tasks.

Methods	5-way	
	1-shot	5-shot
MANN, no conv [10]	82.8%	94.9%
MAML, no conv [2]	89.7 \pm 1.1%	97.5 \pm 0.6 % (96.1 \pm 0.4)%*
TAML(Entropy), no conv	91.19 \pm 1.03%	97.40 \pm 0.34%
TAML(Theil), no conv	91.37 \pm 0.97%	96.84 \pm 0.36%
TAML(GE(2)), no conv	91.3 \pm 1.0%	96.76 \pm 0.4%
TAML(Atkinson), no conv	91.77 \pm 0.97%	97.0 \pm 0.4%
Siamese Nets [12]	97.3%	98.4%
Matching Nets [13]	98.1%	98.9%
Neural Statistician [20]	98.1%	99.5%
Memory Mod. [21]	98.4%	99.6%
Prototypical Nets [14]	98.8%	99.7%
Meta Nets [11]	98.9%	-
Snail [9]	99.07 \pm 0.16%	99.78 \pm 0.09%
MAML [2]	98.7 \pm 0.4%	99.9 \pm 0.1%
MAML+L2 [2]	98.77 \pm 0.5%	99.31 \pm 0.1%
Meta-SGD* [8]	97.97 \pm 0.7%	98.96 \pm 0.2%
TAML(Entropy)	99.23 \pm 0.35%	99.71 \pm 0.1%
TAML(Theil)	99.5 \pm 0.3%	99.81 \pm 0.1%
TAML(GE(2))	99.47 \pm 0.25%	99.83 \pm 0.09%
TAML(Atkinson)	99.37 \pm 0.3%	99.77 \pm 0.1%
TAML (Gini-Coefficient)	99.3 \pm 0.32%	99.70 \pm 0.1%
TAML(GE(0))	99.33 \pm 0.31%	99.75 \pm 0.09%
TAML (VL)	99.1 \pm 0.36%	99.6 \pm 0.1%

Table 2. Few Shot Classification results on Omniglot dataset for CNN on 20-way setting. For a fair comparison, * denotes re-run results by both meta-learning approaches on the same training/test split used in TAML models. The proposed TAML approaches outperform both MAML and Meta-SGD.

Methods	20-way	
	1-shot	5-shot
Siamese Nets [12]	88.2%	97.0%
Matching Nets [13]	93.8%	98.5%
Neural Statistician [20]	93.2%	98.1%
Memory Mod. [21]	95.0%	98.6%
MAML* [2]	90.81 \pm 0.5%	97.49 \pm 0.15%
MAML+L2* [2]	90.93 \pm 0.6%	97.65 \pm 0.18%
Meta-SGD* [8]	93.98 \pm 0.43%	98.42 \pm 0.11%
TAML(Entropy + MAML)	95.62 \pm 0.5%	98.64 \pm 0.13%
TAML(Theil + Meta-SGD)	95.15 \pm 0.39%	98.56 \pm 0.1%
TAML(Atkinson + Meta-SGD)	94.91 \pm 0.42%	98.50 \pm 0.1%
TAML (VL + Meta-SGD)	95.12 \pm 0.39%	98.58 \pm 0.1%
TAML(Theil + MAML)	92.61 \pm 0.46%	98.4 \pm 0.1%
TAML(GE(2) + MAML)	91.78 \pm 0.5%	97.93 \pm 0.1%
TAML(Atkinson + MAML)	93.01 \pm 0.47%	98.21 \pm 0.1%
TAML(GE(0) + MAML)	92.95 \pm 0.5%	98.2 \pm 0.1%
TAML (VL + MAML)	93.38 \pm 0.47%	98.54 \pm 0.1%

4.1. Classification

We use two benchmark datasets Omniglot and MiniImagenet for few-shot classification problem. The Omniglot dataset has 1623 characters from 50 alphabets. Each charac-

ter has 20 instances which are drawn by different individuals. We randomly select 1200 characters for training and remaining for testing. From 1200 characters, we randomly sample 100 for validation. As proposed in [10], the dataset is augmented with rotations by multiple of 90 degrees.

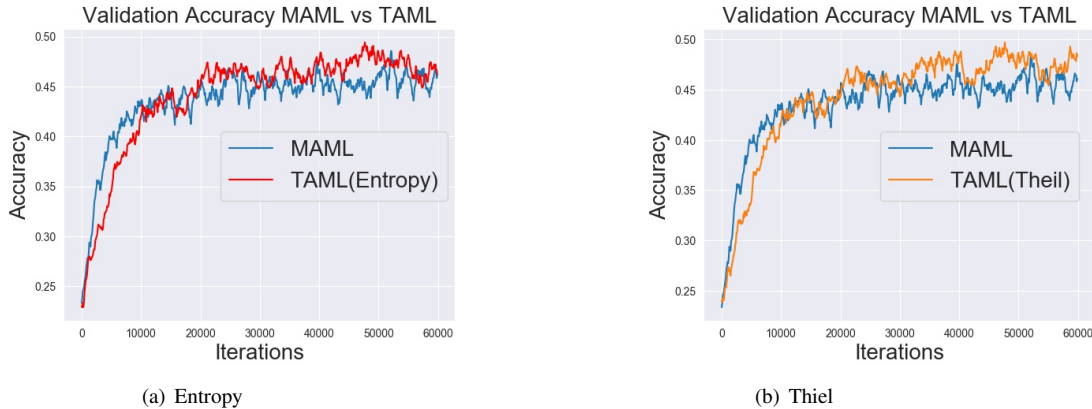


Figure 1. Validation Accuracy of TAML vs MAML on Mini-Imagenet 1-shot 5-way.

Algorithm 2 Inequality Measures Based TAML for Reinforcement Learning

Require: $p(\mathcal{T})$: distribution over tasks.

Require: α, β : hyperparameters

Randomly Initialize θ

while not done **do**

 Sample batch of tasks $\mathcal{T}_i \sim p(\mathcal{T})$

for all \mathcal{T}_i **do**

 Sample K trajectories (x_1, a_1, \dots, x_T) using f_θ in \mathcal{T}_i .

 Evaluate $\nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$ and $\mathcal{L}_{\mathcal{T}_i}$ using K trajectories in Equation 7

 Compute adapted parameters using gradient descent : $\theta_i = \theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_i}$.

 Sample trajectories $\mathcal{D}_{val,i}$ using f_{θ_i} in \mathcal{T}_i .

end for

 Update $\theta \leftarrow \theta - \beta \nabla_\theta [\mathbb{E}_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta_i}) + \lambda \mathcal{I}_{\mathcal{E}}(\{\mathcal{L}_{\mathcal{T}_i}(f_\theta)\})]$ using $\mathcal{D}_{val,i}$, $\mathcal{L}_{\mathcal{T}_i}$, and $\mathcal{I}_{\mathcal{E}}$

end while

The Mini-Imagenet dataset was proposed by [13] and it consists of 100 classes from Imagenet dataset. We used the same split proposed by [1] for fair comparison. It involves 64 training classes, 12 validation classes and 20 test classes. We consider 5-way and 20-way classification for both 1-shot and 5-shot.

For K -shot N -way classification, we first sample N unseen classes from training set and for every N unseen class, we sample K different instances. We follow the same model architecture used by [13]. The Omniglot dataset images are downsampled by 28x28 and we use a strided convolutions instead of max-pooling. The MiniImagenet images are downsampled to 84x84 and we used 32 filters in the convolutional layers for 5-shot setting. For 1-shot setting, we used 64 filters in convolutional layers and we added two dropouts. We also used Leaky-ReLU instead of ReLU as non-linearity. We

re-run the MAML on MiniImagenet 1-shot setting for this customized architecture too. We also evaluate the proposed approach on non-convolutional neural network. For a fair comparison with MANN [10] and MAML [2], we follow the same architecture used by MAML [2]. We use Leaky-ReLU as non-linearity instead of ReLU non-linearity.

We train and evaluate the meta-models based on TAML that are unbiased and show they can be adapted to new tasks in few iterations as how they are meta-trained. For Omniglot dataset, we use a batch size of 32 and 16 for 5-way and 20-way classification, respectively. We follow [2] for other training settings. For fair comparison with Meta-SGD on 20-way classification, the model was trained with 1 gradient step. For 5-way Mini-Imagenet, we use a batch size of 4 for both 1-shot and 5-shot settings. For 5-way 5-shot setting, we used a learning rate α of 0.05. For 20-way classification on MiniImagenet, the learning rate was set to 0.01 for both 1-shot and 5-shot, and each task is updated using one-gradient step. All the models are trained for 60000 iterations. We use the validation set to tune the hyper-parameter λ for both the approaches.

4.1.1 Results

We report the results for 5-way Omniglot for both fully connected network and convolutional network. We added one more baseline in which we add L2 regularizer in the MAML’s cost function and from Table 1 2, it shows that the performance is about the same as MAML for both 5-way and 20-way classification settings. The convolutional network learned by TAML outperforms all the state-of-the-art methods in Table 1. For 20-way classification, we re-ran the Meta-SGD algorithm with our own training/test splitting for fair comparison since the Meta-SGD is not open-sourced and their training/test split is neither available. The results are reported in the Table 2. It can be shown that TAML outperforms MAML and Meta-SGD for both 1-shot and 5-

Table 3. Few Shot Classification results on Mini-Imagenet dataset on 5-way and 20-way setting. The results for other methods on 5-way are reported from MAML, and for 20-way, the results are reported from Meta-SGD. TAML approaches outperform MAML on both settings and Meta-SGD on 20-way setting.* Accuracy using the comparable network architecture.

Methods	5-way		20 way	
	1-shot	5-shot	1-shot	5-shot
Fine-tune	28.86 ± 0.54%	49.79 ± 0.79%	-	-
Nearest Neighbors	41.08 ± 0.70%	51.04 ± 0.65%	-	-
Matching Nets [13]	43.56 ± 0.84%	55.31 ± 0.73%	17.31 ± 0.22%	22.69 ± 0.20%
Meta-Learn LSTM [1]	43.44 ± 0.77%	60.60 ± 0.71%	16.70 ± 0.23%	26.06 ± 0.25%
TAML(Theil + Meta-Learn LSTM)	46.28 ± 0.79%	62.92 ± 0.66%	-	-
MAML (firstorderapprox.) [2]	48.07 ± 1.75%	63.15 ± 0.91%	-	-
MAML [2]	48.70 ± 1.84%	63.11 ± 0.92%	16.49 ± 0.58%	19.29 ± 0.29%
MAML (64 filters) [2]	49.5 ± 1.8%	-	-	-
Meta-SGD [8]	50.47 ± 1.87%	64.03 ± 0.94%	17.56 ± 0.64%	28.92 ± 0.35%
Prototypical network [14]	46.61 ± 0.78%	65.77 ± 0.70%	-	-
Reptile [22]	49.97 ± 0.32%	65.99 ± 0.58%	-	-
LLAMA [23]	49.40 ± 1.83%	-	-	-
SNAIL* [9]	45.1%	55.2%	-	-
GNN [24]	50.33 ± 0.36%	66.41 ± 0.63%	-	-
Relation Network [25]	50.44 ± 0.82%	65.32 ± 0.70%	-	-
TAML(Entropy + MAML)	51.73 ± 1.88%	66.05 ± 0.85%	-	-
TAML(Theil + MAML)	51.5 ± 1.86%	65.94 ± 0.9%	18.74 ± 0.65%	25.77 ± 0.33%
TAML(GE(2) + MAML)	50.87 ± 1.86%	65.18 ± 0.9%	18.22 ± 0.67%	24.89 ± 0.34%
TAML(Atkinson + MAML)	51.03 ± 1.83%	65.24 ± 0.91%	-	-
TAML(GE(0) + MAML)	50.93 ± 1.9%	65.71 ± 0.9%	18.95 ± 0.68%	24.53 ± 0.33%
TAML (VL + MAML)	51.13 ± 1.85%	66.0 ± 0.89%	18.13 ± 0.64%	25.33 ± 0.32%
TAML(GE(0) + Meta-SGD)	51.1 ± 1.88%	65.51 ± 0.93%	19.45 ± 0.67%	29.75 ± 0.34%
TAML (VL + Meta-SGD)	51.77 ± 1.86%	65.6 ± 0.93%	19.73 ± 0.65%	29.81 ± 0.35%

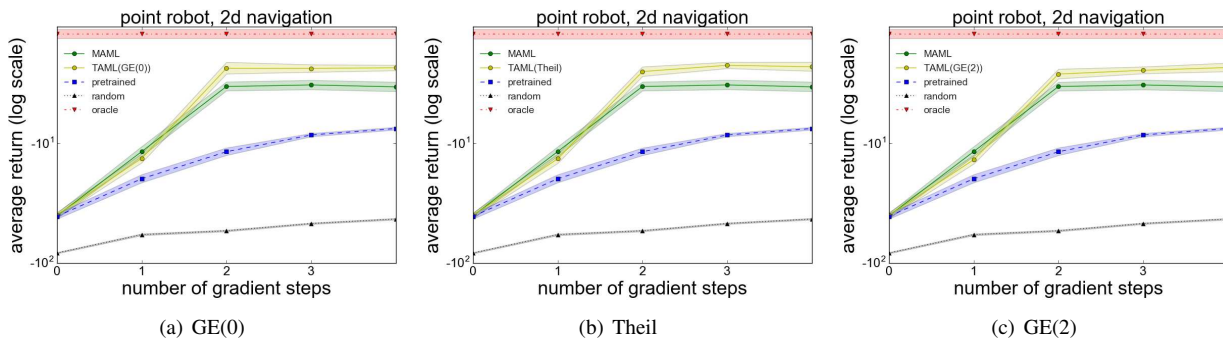


Figure 2. Results on 2D Navigation task.

shot settings. The results also show that TAML achieves much more competitive rule during the training.

For MiniImagenet, the proposed TAML approaches outperform the compared ones for 1-shot 5-way classification problem. For 5-shot 5 way setting, our entropy based approach still outperforms all the other methods except for GNN which is still within the variance of entropy based approach. The entropy based TAML achieves the best per-

formance compared with inequality-minimization TAML for 5-shot problem. For 20-way setting, we use the reported results from Meta-SGD for both MAML and Meta-SGD. We outperform both MAML and Meta-SGD for both 1-shot and 5-shot settings. It is interesting to note that MAML performs poor compared with matching nets and Meta-learner LSTM when it is trained using one gradient step as reported in Table 3. The test accuracy for prototypical results which is

reported in Table 3 for models matches train and test "shot" and "way". The results reported by [14] requires 30-way 15 queries per training episode for 1-shot and, 20-way 15 queries per training episode for 5-shot results.

We also compare the performance of TAML when applied to Meta-Learn LSTM [1]. For this experiment, we added dropout after the last convolution layer and use leaky ReLU instead of ReLU non-linearity. In each iteration, We sample 5 datasets where each dataset is $\{\mathcal{D}_{train}, \mathcal{D}_{test}\}$ from $\mathcal{D}_{meta-train}$, and then calculate loss for each test set \mathcal{D}_{test} of the dataset using the initial parameter of Meta-Learner. We optimize the parameters of the Meta-learner based on the both classification loss and TAML based inequality measure. We report the result in table 3. For both 1-shot and 5-shot experiment, we outperform Meta-Learn LSTM and achieve almost more than 3% accuracy on both the settings. This shows that TAML can be applicable to any meta-learning algorithm.

Figure 1 shows the curve of validation accuracies of our entropy approach on the left panel and Theil based approach on right panel versus MAML for Mini-Imagenet 5-way 1-shot at gradient step 5. It can be seen that our both approaches achieve much better validation accuracy as compared to MAML meaning TAML achieves much better initialization point.

4.1.2 Analysis

Entropy based approach performs better than the inequality based approach. For 5- way Omniglot, there is negligible difference between the entropy based approach and inequality based approach. For 1 shot 5-way MiniImagenet experiment, entropy based TAML still beats inequality based TAML for MAML algorithms. VL based TAML has negligible improvement as compare to entropy based TAML because it uses Meta-SGD algorithm. When it uses MAML, its performance is lower than the entropy based approach. Every inequality has some properties as mentioned in section 3.2. Some of the inequalities are more sensitive to upper part of the distribution means it is more sensitive towards those tasks which have higher loss value and some of the inequalities are sensitive towards changes to those tasks which have lower loss values. The idea is to increase the uncertainty of the initial model on different tasks. Theil inequality is a part of larger family of GE. When alpha is 1 in equation 3, it becomes Theil Index. Moreover, As we can see from Table 1 2 3, VL, GE(0) and Thiel perform better than GE(2). For Omniglot 5-way experiment, the margin is negligible because MAML already achieved 99% accuracy on 1 shot and 99.9% on 5-shot. Atkinson index can also be derived from generalized entropy index family by setting epsilon = 1 - alpha as mentioned in equation 4 and 3. The high epsilon corresponds to GE index with small alpha means it becomes

sensitive to lower end of the distribution.

4.2. Reinforcement Learning

In reinforcement learning, the goal is to learn the optimal policy given fewer trajectories or experiences. A reinforcement learning task \mathcal{T}_i is defined as Markov Decision Process that consists of a state space \mathcal{S} , an action space \mathcal{A} , the reward function \mathcal{R} , and state-transition probabilities $q_i(x_{t+1}|x_t, a_t)$ where a_t is the action at time step t [26, 27]. In our experiments, we are using the same settings as proposed in [2] where we are sampling trajectories using policy f_θ . The loss function used is the negative of the expectation of the sum of the rewards,

$$\mathcal{L}_{\mathcal{T}_i} = -\mathbb{E}_{a_t \sim f_\theta, x_t, q_{\mathcal{T}_i}} \left(\sum_{t=1}^T \mathcal{R}_i(x_t, a_t) \right). \quad (7)$$

Experiments were performed using rllab suite [28]. Vanilla policy gradient [29] is used for inner gradient updates while trust region policy optimizer (TRPO) [30] is used as meta-optimizer. The algorithm is the same as mentioned in algorithm 1 with the only difference being that trajectories were sampled instead of images.

For reinforcement learning experiment, we evaluate TAML on a 2D navigation task. The policy network that was used in performing this task is identical to the policy network that was used in [2] for a fair comparison, which is a three-layered network using ReLU while setting the step size $\alpha = 0.1$. The experiment consists an agent moving in two-dimensional environment and the goal of the agent is to reach the goal state that is randomly sampled from a unit square. For evaluation purposes, we compare the results of TAML with MAML, oracle policy, conventional pre-training and random initialization. Our results have shown that GE(0), Theil, and GE(2) TAML perform on-par with MAML after 2 gradient steps but start to outperform it afterwards as shown in figure 2.

5. Conclusion

In this paper, we proposed a novel paradigm of Task-Agnostic Meta-Learning (TAML) algorithms to train a meta-learner unbiased towards a variety of tasks before its initial model is adapted to unseen tasks. Both an entropy-based TAML and a general inequality-minimization TAML applicable to more ubiquitous scenarios are presented. We argue that the meta-learner with unbiased task-agnostic prior could be more generalizable to handle new tasks compared with the conventional meta-learning algorithms. The experiment results also demonstrate the TAML could consistently outperform existing meta-learning algorithms on both few-shot classification and reinforcement learning tasks.

References

- [1] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *In International Conference on Learning Representations (ICLR)*, 2017. 1, 2, 6, 7, 8
- [2] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *CoRR*, abs/1703.03400, 2017. 1, 2, 5, 6, 7, 8
- [3] Jurgen Schmidhuber. Evolutionary principles in self-referential learning. on learning now to learn: The meta-meta-meta...-hook. Diploma thesis, Technische Universitat Munchen, Germany, 14 May 1987. 2
- [4] D. K. Naik and R. J. Mammone. Meta-neural networks that learn by learning. In *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks*, volume 1, pages 437–442 vol.1, Jun 1992. 2
- [5] Sebastian Thrun and Lorien Pratt, editors. *Learning to Learn*. Kluwer Academic Publishers, Norwell, MA, USA, 1998. 2
- [6] Ke Li and Jitendra Malik. Learning to optimize. *CoRR*, abs/1606.01885, 2016. 2
- [7] Marcin Andrychowicz, Misha Denil, Sergio Gomez Colmenarejo, Matthew W. Hoffman, David Pfau, Tom Schaul, and Nando de Freitas. Learning to learn by gradient descent by gradient descent. *CoRR*, abs/1606.04474, 2016. 2
- [8] Zhenguang Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-sgd: Learning to learn quickly for few shot learning. *CoRR*, abs/1707.09835, 2017. 2, 5, 7
- [9] Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. A simple neural attentive meta-learner. In *International Conference on Learning Representations*, 2018. 2, 5, 7
- [10] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy P. Lillicrap. One-shot learning with memory-augmented neural networks. *CoRR*, abs/1605.06065, 2016. 2, 5, 6
- [11] Tsendsuren Munkhdalai and Hong Yu. Meta networks. *CoRR*, abs/1703.00837, 2017. 2, 5
- [12] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. 2015. 2, 5
- [13] Oriol Vinyals, Charles Blundell, Timothy P. Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. *CoRR*, abs/1606.04080, 2016. 2, 5, 6, 7
- [14] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, 2017. 2, 5, 7, 8
- [15] H. Theil. *Economics and information theory*. Studies in mathematical and managerial economics. North-Holland Pub. Co., 1967. 4
- [16] Frank A. Cowell. Generalized entropy and the measurement of distributional change. *European Economic Review*, 1980. 4
- [17] Anthony B Atkinson. On the measurement of inequality. *Journal of Economic Theory*, 1970. 4
- [18] Paul D. Allison. Measures of inequality. *American Sociological Review*, 1978. 4
- [19] Efe A. Ok and James Foster. Lorenz Dominance and the Variance of Logarithms. Technical report, C.V. Starr Center for Applied Economics, New York University, 1997. 4
- [20] Harrison Edwards and Amos Storkey. *Towards a Neural Statistician*. 2 2017. 5
- [21] Lukasz Kaiser, Ofir Nachum, Aurko Roy, and Samy Bengio. Learning to remember rare events. *CoRR*, abs/1703.03129, 2017. 5
- [22] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *CoRR*, abs/1803.02999, 2018. 7
- [23] Erin Grant, Chelsea Finn, Sergey Levine, Trevor Darrell, and Thomas Griffiths. Recasting gradient-based meta-learning as hierarchical bayes. In *International Conference on Learning Representations*, 2018. 7
- [24] Victor Garcia Satorras and Joan Bruna Estrach. Few-shot learning with graph neural networks. In *International Conference on Learning Representations*, 2018. 7
- [25] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H. S. Torr, and Timothy M. Hospedales. Learning to compare: Relation network for few-shot learning. *CoRR*, abs/1711.06025, 2017. 7
- [26] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fiedjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharmashan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, February 2015. 8
- [27] H. Sheikh and L. Bölöni. Emergence of scenario-appropriate collaborative behaviors for teams of robotic bodyguards. In *to be presented at Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS-2019)*, May 2019. 8
- [28] Yan Duan, Xi Chen, Rein Houthoofd, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. *CoRR*, abs/1604.06778, 2016. 8
- [29] Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3):229–256, May 1992. 8
- [30] John Schulman, Sergey Levine, Philipp Moritz, Michael I. Jordan, and Pieter Abbeel. Trust region policy optimization. *CoRR*, abs/1502.05477, 2015. 8