Task Similarity Aware Meta Learning: Theory-inspired Improvement on MAML

Pan Zhou[*] **Yingtian Zou**[†] **Xiao-Tong Yuan**[‡] **Jiashi Feng**[†] **Caiming Xiong**[*] **Steven Hoi**[*]

[*]Salesforce Research [†]National University of Singapore [‡]Nanjing University of Information Science & Technology
panzhou3@gmail.com zouyingt@comp.nus.edu.sg xtyuan@nuist.edu.cn
elefjia@nus.edu.sg {cxiong,shoi}@salesforce.com

Abstract

Few-shot learning ability is heavily desired for machine intelligence. By meta-learning a model initialization from training tasks with fast adaptation ability to new tasks, model-agnostic meta-learning (MAML) has achieved remarkable success in a number of few-shot learning applications. However, theoretical understandings on the learning ability of MAML remain absent yet, hindering developing new and more advanced meta learning methods in a principled way. In this work, we solve this problem by theoretically justifying the fast adaptation capability of MAML when applied to new tasks. Specifically, we prove that the learnt meta-initialization can benefit the fast adaptation to new tasks with only a few steps of gradient descent. This result explicitly reveals the benefits of the unique designs in MAML. Then we propose a theory-inspired task similarity aware MAML which clusters tasks into multiple groups according to the estimated optimal model parameters and learns group-specific initializations. The proposed method improves upon MAML by speeding up the adaptation and giving stronger few-shot learning ability. Experimental results on the few-shot classification tasks testify its advantages.

1 INTRODUCTION

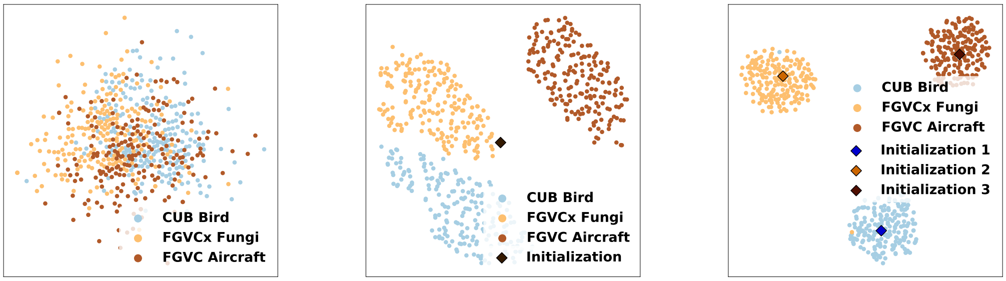
Meta learning [Schmidhuber, 1987, Naik and Mammone, 1992, Bengio et al., 1990], a.k.a. learning-to-learn [Thrun and Pratt, 2012], offers a new way to solve few-shot learning tasks via learning task-level knowledge. Specifically, at task level it trains a meta learner to extract task-shared knowledge from all the training tasks; then the meta learner is used to facilitate a task-specific model to learn a new task with only a small amount of data [Ravi and Larochelle, 2017, Finn et al., 2017, Santoro et al., 2016, Vinyals et al., 2016, Nichol and Schulman, 2018]. Among existing meta learning methods, model-agnostic meta-learning (MAML) [Finn et al., 2017] is a representative one because of its simplicity, generality and state-of-the-art performance [Nichol and Schulman, 2018, Antoniou et al., 2019, Li et al., 2017]. It aims to learn a meta model from the observed tasks that could serve as a good initialization for task-specific models. Then given a test task, it only applies a few gradient descent steps on a few training samples for adapting the meta model to the test task, since the learnt initial model is desired to be close to the optimal models of the observed tasks and thus can be quickly adapted to new similar tasks.

Despite its remarkable success in practice [Finn et al., 2017, Duan et al., 2016, Li et al., 2017], the theoretical understanding of MAML is still largely absent. Specifically, it is not clear *why MAML is able to generalize well in new tasks via merely taking a few steps of gradient descent on a small amount of data.* The answer to this question is important not only for justifying the fast adaptation capability of MAML, but also for inspiring new insights for algorithm improvement.

Contributions. In this work, we address the above fundamental question and contribute to derive some new results, insights and alternatives for MAML. Particularly, we provide rigorous theoretical analysis for its generalization behaviors. Inspired by our theory, we then propose a new alternative of MAML which is more effective for few-shot learning. Our main contributions are highlighted below.

Our first contribution is proving that in MAML, its learnt meta-initialization can benefit the fast adaptation to new tasks with only a few steps of gradient descent. Specifically, let θ^* be the initialization learnt by MAML with meta model $f(\theta, x)$ on the training tasks which are drawn from a task distribution $\mathcal{T}$. For a task T, let $\mathcal{L}_{D_T}(\theta) = \frac{1}{K}\sum_{(x,y)\in D_T}\ell(f(\theta,x),y)$ denote its empirical risk on its training dataset D_T of size K. Then for any test task $T \sim \mathcal{T}$, we prove that its task-specific adapted parame-

Accepted for the 37th Conference on Uncertainty in Artificial Intelligence (UAI 2021).



(a) (b) (c)

Figure 1: Illustration of the learnt group structures by MAML and TSA-MAML on 5-shot 5-way learning task of a group-structured dataset with three sub-datasets, i.e. Aircraft [Maji et al., 2013], CUB Birds [Wah et al., 2011] and FGVCx Fungi [Maji et al., 2018]. One can observe indistinguishable sample features of tasks in (a) but well group-structured optimal model parameters of tasks learnt by MAML and TSA-MAML via 10 gradient descent steps from learnt initializations in (b) and (c) respectively. See details in Sec. 5.1.

ter $\theta_T^q = \theta^* - \alpha\big[\nabla\mathcal{L}_{D_T}(\theta^*) + \sum_{t=1}^{q-1}\nabla\mathcal{L}_{D_T}(\theta_T^t)\big]$ obtained by taking q gradient descent steps on its training data D_T has good performance on its test data $(x, y) \sim T$, where $\theta_T^1 = \theta^* - \alpha\nabla\mathcal{L}_{D_T}(\theta^*)$. Specifically, by defining population risk $\mathcal{L}(\theta) = \mathbb{E}_{(x,y)\sim T}\ell(f(\theta, x), y)$ on task T, we show the excess risk $\mathbb{E}_{T\sim\mathcal{T}}\mathbb{E}_{D_T}\left[\mathcal{L}(\theta_T^q) - \mathcal{L}(\theta_T^*)\right]$ of θ_T^q, well measuring the testing performance, is upper bounded by $\mathcal{O}\big(\frac{\rho^q}{K} + \mathbb{E}_{T\sim\mathcal{T}}\mathbb{E}_{D_T}\big[\mathcal{L}_{D_T}(\theta_T^q) - \mathcal{L}(\theta_T^*)\big]\big)$, where the constant ρ is slightly larger than one, and θ_T^* is the optimum of population risk $\mathcal{L}(\theta)$ on T. This result explicitly reveals the importance of the gradient step number q in MAML. Indeed, it suggests us to adapt the learnt initialization θ^* to new task via a few gradient descent steps. See details in Sec. 3.2. Besides, we further upper bound $\mathbb{E}_{T\sim\mathcal{T}}\mathbb{E}_{D_T}\big[\mathcal{L}_{D_T}(\theta_T^q) - \mathcal{L}(\theta_T^*)\big]$ by $\frac{1}{2\alpha}\mathbb{E}_{T\sim\mathcal{T}}\left[\|\theta^* - \theta_T^*\|_2^2\right]$, showing the smaller distance between θ^* and θ_T^* the smaller the excess risk. Meanwhile, as the learnt initialization θ^* by MAML is often close to θ_T^*, our results can explain why MAML generalizes well to new tasks to some extent.

Inspired by our theory, we further develop the task similarity aware MAML (TSA-MAML) as a novel alternative to achieve faster adaptation to new tasks. As shown in Fig. 1 (a) and (b), though the samples in tasks are undistinguishable, the optimal model parameters estimated by MAML have remarkable group structures. So instead of learning one initialization for all tasks, TSA-MAML leverages task similarity to discover the group structures in the tasks by using a learner $\mathcal{A}$ to measure task similarity in terms of the estimated task-specific model parameters. Then to facilitate the learning of new tasks, it learns multiple model initializations each of which corresponds to a group of similar tasks. Specifically, given a training task, TSA-MAML first uses the learner $\mathcal{A}$ to predict its group membership and assign a group-specific initialization to it for few-shot training. Next, the initializations are in turn improved and become more group-specific. Consequently, as shown in Fig. 1 (c), the optimal model parameters of tasks in the same group are much closer to the group-specific initialization learnt by TSA-MAML than one common initialization learnt for all tasks by MAML. So TSA-MAML can adapt to new tasks more quickly and better under the few-shot learning setting. In this work, we implement the learner $\mathcal{A}$ as the vanilla MAML and measure the task similarity according to the Euclidean distance between task-specific model parameters. We also theoretically show the superiority of TSA-MAML over MAML on learning new tasks. Extensive experimental results also well demonstrate the advantages of our approach on the few-shot learning problems.

2 RELATED WORK

Meta learning has gained much attention recently because of its success in many applications [Finn et al., 2017, Duan et al., 2016, Mishra et al., 2017, Sung et al., 2017, Xiao et al., 2021, Lin et al., 2021, Zhou et al., 2019, Bai et al., 2021]. The current methods can be divided as metric-based family [Koch et al., 2015, Vinyals et al., 2016, Sung et al., 2018, Snell et al., 2017] that learns sample similarity metrics, memory-based family [Weston et al., 2014, Santoro et al., 2016, Munkhdalai and Yu, 2017] that learns a fast adaptation algorithm via memory models [Graves et al., 2014], and optimization-based family [Finn et al., 2017, Ravi and Larochelle, 2017, Li et al., 2017, Nichol and Schulman, 2018] that learns a model initialization for fast adaptation. Among them, optimization based methods are more preferable, thanks to its simplicity and effectiveness [Finn et al., 2017, Antoniou et al., 2019, Khodak et al., 2019]. One representative method in this line is MAML [Finn et al., 2017] that learns a network initialization such that the network can adapt to a new task via a few gradient descent steps. Later, various variants are proposed to improve MAML [Finn et al., 2019, Li et al., 2017, Yao et al., 2019, **?**]. Among them, HSML [Yao et al., 2019] considers the hierarchical parameter structures in tasks by learning task embeddings to measure task similarity. But it has two issues: (1) feature similarity cannot well reveal model parameter structures in tasks as shown in Fig. 1 and (2) learning similar embeddings for similar tasks is hard, as one cannot well align sample orders in tasks without global sample information (labels) and recurrent networks is sensitive to input orders. In contrast, we measure task similarity in the model parameter space and avoid the above issues. To handle multimodal task distribution, for a task T, MMAML [Vuorio et al., 2019] first learns its task embedding and then its task-specific parameter τ which modulates meta-initialization θ as inner-product initialization $\tau \odot \theta$ for T. It does not explicitly utilize task similarity as it still learns task-specific initialization. In contrast, we explore task structure by clustering similar tasks and learn group-specific initialization. Moreover, like [Yao et al., 2019], learning similar embeddings for similar tasks is hard. Besides, MMAML needs accurate task-specific parameter τ to align with high-dimensional θ to obtain accurate task initialization, increasing learning difficulty. TSA-MAML also differs from multi-task learning, *e.g.* [Kang et al., 2011, Kumar and Daume, 2013, Pentina et al., 2015], as TSA-MAML learns group-specific initialization with fast adaptation ability to new tasks, while the later directly learns task-specific optimal model.

The theoretical analysis of MAML is rarely investigated though heavily desired. Golmant [2019] and Finn et al. [2019] showed the convergence of MAML under strongly convex setting. In [Fallah et al., 2019, Ji et al., 2020], the convergence behavior of MAML on non-convex problems were studied. Saunshi et al. [2020] analyzed the sample complexity for Reptile-alike algorithm [Nichol and Schulman, 2018] instead of MAML. The works [Baxter, 2000, Maurer, 2005, Amit and Meir, 2018, Mikhail et al., 2019, Du et al., 2020, Bai et al., 2021] study the generalization performance of meta learning. But they focus on general meta learning methods and their results do not well reveal any unique property of MAML. For instance, they cannot explain why a few gradient descent steps on a few data in MAML is sufficient to obtain good testing performance. In contrast, by focusing on MAML itself, our theory well justifies this essential design in MAML. Besides, our results are more heuristic and directly derive a new MAML variant which leverages task similarity to facilitate new task learning and is well testified by experimental results.

3 THEORETICAL ANALYSIS OF MAML

Here we first briefly recall the formulation of MAML and then analyze the testing performance of its adapted task-specific model via a few gradient descent steps.

3.1 FORMULATION OF MAML

MAML [Finn et al., 2017] is to learn a good initialization parameter θ for a class of parameterized learner $f : \mathcal{X} \mapsto \mathcal{Y}$ (*e.g.* a classifier) such that for any task T drawn from a task distribution $\mathcal{T}$, its task-specific adapted parameter θ_T via one gradient descent step from θ on a small training dataset $D_T^{tr} = \{(x_i, y_i)\}_{i=1}^K$ can perform well on its test dataset $D_T^{ts} = \{(\widetilde{\boldsymbol{x}}_i, \widetilde{\boldsymbol{y}}_i)\}_{i=1}^K$. Towards this goal, for each task $T \sim \mathcal{T}$, MAML optimizes the test loss of its adapted parameter θ_T as follows

$$\min_\theta \mathbb{E}_{T\sim\mathcal{T}}\, \mathcal{L}_{D_T^{ts}}(\theta - \alpha\nabla\mathcal{L}_{D_T^{tr}}(\theta)),$$

where $\mathcal{L}_{D_T}(\theta_T) = \frac{1}{K}\sum_{(x,y)\in D_T}\ell(f(\theta_T, x), y)$ with $D_T = D_T^{tr}$ or D_T^{ts} is the empirical risk on the dataset D_T, and α is a learning rate. Here the function $\ell(f(\theta_T, x), y)$ measures the discrepancy between the prediction $f(\theta_T, x)$ and the ground truth y, *e.g.* the cross-entropy loss in classification.

After learning the initialization θ^*, given a test task $T \sim \mathcal{T}$ with small training and test datasets D_T^{tr} and D_T^{ts} respectively, MAML adapts θ^* to task T via a few gradient descent steps on D_T^{tr} and then tests the adapted parameter on D_T^{ts}. In spite of its impressive performance, there is no rigorous theoretical analysis of MAML that explicitly justifies effectiveness of a few gradient based adaptation. The following sections attempt to solve this issue by developing testing performance guarantees.

3.2 TESTING PERFORMANCE ANALYSIS

Here we answer two questions: (1) what factors affect the test performance of the adapted model in MAML via a few gradient descent adaptation steps on a few training data; (2) how the learnt initialization benefits the learning of future tasks. Let $T \sim \mathcal{T}$ be any future task with K training samples $D_T = \{(x_i, y_i)\}_{i=1}^K$. Assume we run q gradient descent steps on the data D_T to obtain the adapted model $\theta_T^q = \theta^* - \alpha[\nabla\mathcal{L}_{D_T}(\theta^*) + \sum_{t=1}^{q-1}\nabla\mathcal{L}_{D_T}(\theta_T^t)]$ for task T with learnt initialization θ^* and $\theta_T^1 = \theta^* - \alpha\nabla\mathcal{L}_{D_T}(\theta^*)$. Let $\theta_T^* \in \mathrm{argmin}_{\theta_T}\{\mathcal{L}(\theta_T) := \mathbb{E}_{(x,y)\sim T}\,[\ell(f(\theta_T, x), y)]\}$ trained on all samples $(x, y) \sim T$ denote the optimal model parameter of the task $T \sim \mathcal{T}$. Before analysis, we first give necessary definitions which are fairly standard in the optimization analysis of deep network and MAML [Zhou and Feng, 2018b,a, Zhou et al., 2020a,b, Finn et al., 2019, Golmant, 2019, Fallah et al., 2019, Ji et al., 2020, Wu et al., 2020].

Definition 1 (Lipschitz continuity and smoothness)**.** *We say a function $g(\theta)$ is G-Lipschitz continuous if $\|g(\theta_1) - g(\theta_2)\|_2 \leq G\|\theta_1 - \theta_2\|_2$ with a constant G. $g(\theta)$ is said to be L-smooth if $\|\nabla g(\theta_1) - \nabla g(\theta_2)\|_2 \leq L\|\theta_1 - \theta_2\|_2$ with a constant L.*

Then we formally state our results in Theorem 1 which shows the role of q and the benefits of initialization θ^* on reducing the excess risk $\mathsf{ER}(\theta_T^q) = \mathbb{E}_{T\sim\mathcal{T}}\mathbb{E}_{D_T}\big[\mathcal{L}(\theta_T^q) - \mathcal{L}(\theta_T^*)\big]$. As $\mathsf{ER}(\theta_T^q)$ evaluates the loss difference

$[\ell(f(\theta_T^q, x), y) - \ell(f(\theta_T^*, x), y)]$ on all samples $(x, y) \sim T$ and all tasks $T \sim \mathcal{T}$, it can well measure the testing performance of the adapted parameter θ_T^q. See its proof in Appendix C.2.

Theorem 1. ***(Testing Performance Analysis)*** *Suppose $\ell(f(\theta, x), y)$ is G-Lipschitz continuous w.r.t. the parameter θ. We also assume $\ell(f(\theta, x), y)$ is L_s-smooth w.r.t. θ and α obeys $\alpha \leq \frac{1}{L_s}$. By setting $\rho = 1 + 2\alpha L$, then for any $T \sim \mathcal{T}$ and $D_T = \{(x_i, y_i)\}_{i=1}^K \sim T$, we have*

$$\begin{aligned} \mathit{ER}(\theta_T^q) &\overset{(a)}{\leq} \frac{2G^2(\rho^q - 1)}{KL} + \mathbb{E}_{T\sim\mathcal{T}}\mathbb{E}_{D_T}\big[\mathcal{L}_{D_T}(\theta_T^q) - \mathcal{L}(\theta_T^*)\big] \\ &\overset{(b)}{\leq} \frac{2G^2(\rho^q - 1)}{KL} + \frac{1}{2\alpha}\mathbb{E}_{T\sim\mathcal{T}}\big[\|\theta^* - \theta_T^*\|_2^2\big]. \end{aligned}$$

From the first inequality (a) in Theorem 1, one can observe that the excess risk $\mathsf{ER}(\theta_T^q)$ of the task-specific adapted model θ_T^q for task T is determined by two factors, i.e., the training sample number K for each task and the expected loss distance $\mathbb{E}_{T\sim\mathcal{T}}\mathbb{E}_{D_T}\big[\mathcal{L}_{D_T}(\theta_T^q) - \mathcal{L}(\theta_T^*)\big]$ between the adapted parameter θ_T^q provided by MAML and the optimal model θ_T^* for task T. Obviously, the larger training sample number K is, the smaller the first term in the upper bound is. Besides, the closer θ_T^q is to θ_T^*, the better task-specific parameter θ_T^q with smaller excess risk.

Theorem 1 also provides some insights of the effect of adaptation step number q to the excess risk $\mathsf{ER}(\theta_T^q)$. From the results, one way to reduce the loss $\mathcal{L}_{D_T}(\theta_T^q)$ is to increase the number q of gradient descent steps for adaptation which however increases the first term in the upper bound, as ρ is often slightly larger than one since we often use a small learning rate α. To trade-off the first and second terms, q should not be large. This is because the first term always increases, while as shown in Fig. 2 and in Fig. 6 of Appendix 6, the test loss $\mathbb{E}_{T\sim\mathcal{T}}\mathbb{E}_{D_T}\big[\mathcal{L}_{D_T}(\theta_T^q)\big]$ in the second term $\mathbb{E}_{T\sim\mathcal{T}}\mathbb{E}_{D_T}\big[\mathcal{L}_{D_T}(\theta_T^q) - \mathcal{L}(\theta_T^*)\big]$ decreases very fast at the first a few iterations but increases along more optimization iterations due to over-fitting issue. This observation accords with our theory affirmation. So the bound of excess risk $\mathsf{ER}(\theta_T^q)$ can reveal the important characterization of $\mathsf{ER}(\theta_T^q)$ in practice as observed in Fig. 2. The above results also partially explain why MAML often adapts the learnt initialization θ^* to new tasks via a few gradient descent steps.

Besides, the second inequality (b) in Theorem 1 justifies the benefits of the learnt initialization θ^* to the testing performance. Specifically, Theorem 1 shows the smaller distance between θ^* and θ_T^*, the smaller excess risk. Intuitively, if θ^* is close to θ_T^*, the task-specific adapted parameter θ_T^q would be close to θ_T^*, guaranteeing good testing performance of θ_T^q on its corresponding task $T \sim \mathcal{T}$. Fortunately, empirical results of MAML show that as shown in Fig. 2, a few gradient steps (about 4) from θ^* can provide good performance for test task $T \sim \mathcal{T}$, which indicates the small distance $\|\theta^* - \theta_T^*\|_2^2$.

Then we provide the first-order optimality guarantee in Theorem 2. It shows that the adapted parameter θ_T^q has small expected population gradient $\mathsf{EPG}(\theta_T^q) = \mathbb{E}_{T\sim\mathcal{T}}\big[\big\|\mathbb{E}_{D_T}[\nabla\mathcal{L}(\theta_T^q)]\big\|_2^2\big]$ and thus is close to the desired first-order stationary points of $\mathcal{L}(\theta_T)$.

Theorem 2. ***(First-order Optimality Analysis)*** *With the same assumptions in Theorem 1 and $\rho = 1 + 2\alpha L$, then for any $T \sim \mathcal{T}$ and $D_T = \{(x_i, y_i)\}_{i=1}^K \sim T$, we have*

$$\mathit{EPG}(\theta_T^q) \leq \frac{8G^2(\rho^q - 1)^2}{K^2} + 2\mathbb{E}_{T\sim\mathcal{T}}\mathbb{E}_{D_T}\Big[\big\|\nabla\mathcal{L}_{D_T}(\theta_T^q)\big\|_2^2\Big].$$

See its proof in Appendix C.3. Similar to Theorem 1, Theorem 2 also shows the importance of training sample number K and prefers a small gradient step number q, as large q increases the first term in the upper bound fast but decreases the second term slowly. Besides, Theorem 2 reveals the role of the empirical gradient $\mathbb{E}_{D_T}\big[\big\|\nabla\mathcal{L}_{D_T}(\theta_T^q)\big\|_2^2\big]$ on determining the expected population gradient $\mathsf{EPG}(\theta_T^q)$. Since on a small dataset, when the learnt initialization θ^* is close to the first-order stationary points of the empirical risk $\mathcal{L}_{D_T}(\theta)$ of task $T \sim \mathcal{T}$, then taking a few gradient descent steps already guarantees a very small gradient $\nabla\mathcal{L}_{D_T}(\theta_T^q)$ of the adapted parameter θ_T^q. This means that θ_T^q is very close to a desired stationary point of the population risk $\mathcal{L}(\theta_T)$ and thus can enjoy satisfactory testing performance on the corresponding task T. So a good initialization θ^* can facilitate the learning of new tasks $T \sim \mathcal{T}$ which well explains the success of MAML.

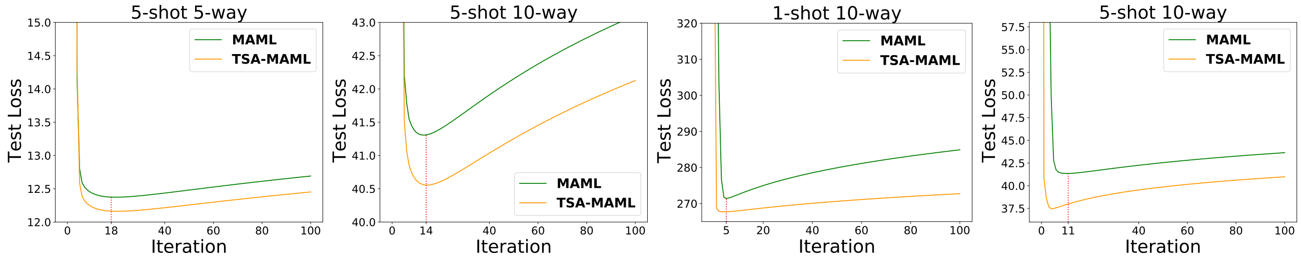
Some works [Baxter, 2000, Maurer, 2005, Amit and Meir, 2018] focused on general meta learning methods and provided generalization performance guarantees which however cannot guarantee the testing performance in this work under fair training performance. Though Mikhail et al. [2019] proved a regret upper bound for a general meta learning framework, their analysis is restricted to online strongly-convex setting and is not applicable to the realistic non-convex settings. Moreover, these aforementioned results do not reveal the unique properties of MAML, *e.g.* the fast adaptation via a few gradient descent steps. Finally, our Theorem 2 provides the first-order optimality guarantee for MAML which is absent in the prior works.

4 TASK SIMILARITY AWARE MAML

Here we introduce the formulation and implementation of our task similarity aware MAML.

4.1 META-PROBLEM FORMULATION

Theorem 1 shows that if one hopes to achieve good testing performance, the learnt initialization θ^* should be close



(a) tieredImageNet (b) miniImageNet

Figure 2: Effects of adaptation step numbers in MAML and TSA-MAML to test loss. See more results (loss and accuracy) in Fig. 6 of Appendix 6. We test on 1,000 tasks and report the average test loss after adaptation. The loss decreases fast at the first a few iterations but increases along more iterations due to over-fitting issue.

to the optimal model parameter θ^*_T of any task $T \sim \mathcal{T}$, i.e. small distance $\mathbb{E}_{T\sim\mathcal{T}}[||\theta^* - \theta^*_T\|_2^2]$. One natural way to further reduce this distance is to learn multiple initializations $\{\theta^*_i\}_{i=1}^m$ and select a correct initialization $\theta^*_{i_T} = \boldsymbol{\mathcal{A}}(\{\theta^*_i\}_{i=1}^m, T)$ from $\{\theta^*_i\}_{i=1}^m$ for a specific task T such that $\mathbb{E}_{T\sim\mathcal{T}}[||\boldsymbol{\mathcal{A}}(\{\theta^*_i\}_{i=1}^m, T) - \theta^*_T\|_2^2]$ is small. Here given a task T, the learner $\boldsymbol{\mathcal{A}}$ assigns it into one of the m groups according to the similarity between T and the tasks in each group such that the optimal model parameter θ^*_T of T is close to the initialization $\boldsymbol{\mathcal{A}}(\{\theta^*_i\}_{i=1}^m, T)$ shared by the tasks in the same group. Here we focus on a general learner $\boldsymbol{\mathcal{A}}$ and provide one effective approach to implement it in Sec. 4.2. Towards this goal, we propose *task similarity aware MAML* (TSA-MAML):

$$\mathbb{E}_{T\sim\mathcal{T}}\mathcal{L}_{D_T^{ts}}(\boldsymbol{\mathcal{A}}(\{\theta_i\}_{i=1}^m, T) - \alpha\nabla\mathcal{L}_{D_T^{tr}}(\boldsymbol{\mathcal{A}}(\{\theta_i\}_{i=1}^m, T)))$$
$$\text{with optimization variables } \{\theta_i\}_{i=1}^m \text{ and } \boldsymbol{\mathcal{A}}.$$

Intuitively, this model aims at using the learner $\boldsymbol{\mathcal{A}}$ to cluster tasks $T \sim \mathcal{T}$ into m groups according to their similarity in terms of their optimal model parameter estimation such that the tasks in each group are sufficiently close to a common initialization. Then based on Theorems 1 and 2, we derive the testing performance bound and first-order optimality of TSA-MAML. Let $\{\theta^*_i\}_{i=1}^m$ be the learnt multiple initializations, $\bar{\theta}^*_T = \boldsymbol{\mathcal{A}}(\{\theta^*_i\}_{i=1}^m, T)$ be the assigned initialization for task T, and θ^q_T be the adapted parameter $\theta^q_T = \bar{\theta}^*_T - \alpha\big[\nabla\mathcal{L}_{D_T}(\bar{\theta}^*_T) + \sum_{t=1}^{q-1}\nabla\mathcal{L}_{D_T}(\theta^t_T)\big]$ for task T with $\theta^1_T = \bar{\theta}^*_T - \alpha\nabla\mathcal{L}_{D_T}(\bar{\theta}^*_T)$. θ^*_T is the optimal model parameter of the population risk $\mathcal{L}(\theta_T) = \mathbb{E}_{(x,y)\sim T}\left[\ell(f(\theta_T, x), y)\right]$ on task T. Then we state our results in Corollary 1 with proof in Appendix D.1.

Corollary 1. *With the same assumptions in Theorem 1 and $\rho = 1 + 2\alpha L$, for any $T \sim \mathcal{T}$ and $D_T = \{(x_i, y_i)\}_{i=1}^K \sim T$, the expected excess risk ER(θ^q_T) obeys*

$$\textit{ER}(\theta^q_T) \leq \frac{2G^2(\rho^q - 1)}{KL} + \frac{1}{2\alpha}\mathbb{E}_{T\sim\mathcal{T}}\big[||\boldsymbol{\mathcal{A}}(\{\theta^*_i\}_{i=1}^m, T) - \theta^*_T\|_2^2\big].$$

Moreover, the population gradient EPG(θ^q_T) obey

$$\textit{EPG}(\theta^q_T) \leq \frac{8G^2(\rho^q - 1)^2}{K^2} + 2\mathbb{E}_{T\sim\mathcal{T}}\mathbb{E}_{D_T}\Big[\big\|\nabla\mathcal{L}_{D_T}(\theta^q_T)\big\|_2^2\Big].$$

Algorithm 1 Meta Framework for TSA-MAML

```
Input: learning rates $\alpha$ and $\beta$, task distribution $\mathcal{T}$.
Initialization: initialize $\{\theta^0_i\}_{i=1}^m$ via the vanilla MAML and $k$-means based approach in Sec. 4.2.
for $t = 0, \cdots, S-1$ do
    sample a task mini-batch $\mathcal{S}^t = \{T_i\}_{i=1}^s$ as $T_i \sim \mathcal{T}$.
    for task $T_i$ in $\mathcal{S}^t$ do
        set initialization $\theta_{i_{T_i}} = \boldsymbol{\mathcal{A}}(\{\theta^t_i\}_{i=1}^m, T_i)$ for $T_i$.
        compute gradient $\nabla\mathcal{L}_{D_T^{tr}}(\theta_{i_{T_i}})$.
        update task-specific parameter $\theta_{T_i}$ as $\theta_{T_i} = \theta_{i_{T_i}} - \alpha\nabla\mathcal{L}_{D_T^{tr}}(\theta_{i_{T_i}})$ for task $T_i$.
    end for
    update $\{\theta^{t+1}_i\}_{i=1}^m$ as follows:
    $\{\theta^{t+1}_i\}_{i=1}^m = \{\theta^t_i\}_{i=1}^m - \beta\sum_{T_i\sim\mathcal{T}}\nabla_{\{\theta^t_i\}_{i=1}^m}\mathcal{L}_{D_{T_i}^{ts}}(\theta_{T_i})$.
end for
Output: $\{\theta^S_i\}_{i=1}^m$
```

Corollary 1 shows that if the learner $\boldsymbol{\mathcal{A}}$ can assign the task $T \sim \mathcal{T}$ into a correct group with a small distance $\mathbb{E}_{T\sim\mathcal{T}}[||\boldsymbol{\mathcal{A}}(\{\theta^*_i\}_{i=1}^m, T) - \theta^*_T\|_2^2]$, TSA-MAML would be expected to have smaller expected excess risk ER(θ^q_T) and thus better testing performance than MAML. This can be intuitively understood: by grouping the tasks $T \sim \mathcal{T}$ into m clusters such that the tasks in the same group have similar optimal model parameters and by learning a group-specific shared initialization for each group, the optimal model parameters of tasks in a group will be much closer to the group-specific shared initialization learnt by TSA-MAML than a common initialization learnt for all tasks $T \sim \mathcal{T}$ in MAML. Accordingly, TSA-MAML requires less samples to adapt to new tasks and thus achieves better testing performance. Moreover, for a task, since the initialization of TSA-MAML is closer to its optimal parameter θ^*_T than that of MAML, after a few gradient descent steps the adapted parameter θ^q_T of TSA-MAML is expected to be closer to θ^*_T, guaranteeing smaller $\mathbb{E}_{T\sim\mathcal{T}}\mathbb{E}_{D_T}\big[\big\|\nabla\mathcal{L}_{D_T}(\theta^q_T)\big\|_2^2\big]$. So θ^q_T in TSA-MAML would have smaller population gradient EPG(θ^q_T), indicating better performance over MAML.

Table 1: Classification accuracy (%) of the compared approaches on the 5-shot 5-way few-shot learning tasks in the two group-structured datasets (600 test episodes with 95% confidence intervals).

	Aircraft + CUB Bird + FGVCx Fungi				Stanford Car + CUB Bird + FGVCx Fungi			
	aircraft	bird	fungi	average	car	bird	fungi	average
Reptile [Nichol and Schulman, 2018]	60.46±0.68	71.96±0.79	51.71±0.84	61.38	43.64±0.64	69.63±0.78	52.06±0.85	55.11
HSML [Yao et al., 2019]	69.89±0.90	68.99±1.01	53.63±1.03	64.17	48.19±0.93	71.20±0.97	53.48±1.08	57.62
MMAML [Vuorio et al., 2019]	56.02±0.63	68.33±0.82	53.44±0.76	59.26	34.97±0.46	64.83±0.80	53.33±0.77	51.04
FOMAML [Finn et al., 2017]	49.60±0.98	69.53±0.95	47.56±0.83	55.56	34.20±0.72	68.50±0.78	46.66±0.89	49.79
MAML [Finn et al., 2017]	67.82±0.65	70.55±0.77	53.20±0.82	63.86	47.67±0.70	68.64±0.82	53.43±0.89	56.25
TSA-MAML	**72.84±0.63**	**74.80±0.76**	**56.86±0.87**	**68.17**	**50.01±0.65**	**73.92±0.80**	**56.03±0.87**	**59.98**

4.2 IMPLEMENTATION AND DISCUSSION

Implementation. The key for implementing TSA-MAML is to design the learner $\mathcal{A}$ which assigns a task T into a correct group such that its optimal model parameter is close to the initialization of the group. Here we implement $\mathcal{A}$ as follows. Firstly, we train vanilla MAML and obtain the initialization θ^* for all tasks $T \sim \mathcal{T}$. Then we use vanilla MAML with initialization θ^* to compute the estimated optimal parameters $\{\bar{\theta}_{T_i}\}_{i=1}^n$ of sufficient sampled tasks $\{T_i\}_{i=1}^n$ and perform k-means [MacQueen, 1967] on $\{\bar{\theta}_{T_i}\}_{i=1}^n$ to cluster them into m groups $\{\mathcal{G}_i\}_{i=1}^m$. See the experimental settings of n and m in Sec. 5.

Next, we initialize each group-specific initialization θ_i^0 by averaging the model parameters $\{\bar{\theta}_{T_i}\}_{i\in\mathcal{G}_i}$. Finally, for training, given a task T, we also first use vanilla MAML with initialization θ^* to compute its estimated optimum $\bar{\theta}_T$, and then find a group $\mathcal{G}_i$ such that the group-specific initialization θ_i has a smallest Euclidean distance to $\bar{\theta}_T$. In this way, we can use task T to update the initialization θ_i for group $\mathcal{G}_i$ like MAML. Note, we measure the task similarity in the model parameter space instead of the task feature space (sample feature) which measures the similarity more accurately, since as shown in Fig. 1, task features cannot well reveal the group structures of the optimal models of tasks and will be discussed in Sec. 5.1 with more details. See detailed algorithm in Algorithm 1.

Discussion. HSML [Yao et al., 2019] considers the hierarchical parameter structures in tasks by learning task embeddings to measure task similarity which however has two issues. (1) Due to complex deep models, feature similarity cannot well reveal the model parameter structures in tasks. For instance, Fig. 1 shows that undistinguishable sample features in tasks still have remarkable group-structured optimal models. (2) Learning similar embeddings for similar tasks is hard, as one cannot align sample orders in tasks without global sample information (labels) and recurrent networks are sensitive to input sample orders. In contrast, we measure task similarity from model parameter space which avoids the above issues and guarantees small distance among optimal models in the same group. To handle multimodal task distribution, MMAML [Vuorio et al., 2019] learns individual embedding for each task T and uses it to learn parameter τ which modulates initialization θ^* as task-specific inner-product initialization $\tau \odot \theta^*$ for T. So it does not explicitly utilize task similarity as it still learns task-specific initialization. Conversely, we explicitly explore task structure by clustering similar tasks, and learn group-specific initializations. Moreover, like HSML, it also faces the issue of learning similar embeddings for similar tasks. Besides, MMAML needs accurate modulation parameter τ to align with high-dimensional θ to produce accurate task-specific initialization, increasing learning difficulty.

5 EXPERIMENTS

Here we compare our TSA-MAML with state-of-the-arts for the few-shot classification tasks.

5.1 EVALUATION ON THE GROUP-STRUCTURED DATA

Datasets. We first investigate whether TSA-MAML can leverage the task similarity to discover task-group structures and further learn group-specific initializations. Towards this goal, we randomly sample each training/test task from one of the three datasets, i.e. Aircraft dataset [Maji et al., 2013], CUB Birds [Wah et al., 2011] and FGVCx-Fungi dataset [Maji et al., 2018]. As each dataset only contains one category, *e.g.* birds, the tasks drawn from each dataset should have similar optimal model parameters, indicating remarkable group structures in these optimal model parameters as illustrated by Fig. 1. Accordingly, discovering these group structures and learning group-specific initializations can benefit new task learning. Similarly, we construct the second group-structured dataset which contains Stanford Car [Krause et al., 2013], CUB Birds [Wah et al., 2011] and FGVCx-Fungi [Maji et al., 2018]. Like conventional setting, each sub-dataset, *e.g.* CUB Birds, contains meta-training, meta-validation and meta-test classes which is specified in [Yao et al., 2019] and Appendix A.

Experimental setting. Following [Finn et al., 2017, Snell et al., 2017], we use the episodic procedure for K-shot N-way few-shot learning task. We use the same 4-layered convolution network in [Finn et al., 2017, Nichol and Schulman, 2018] for evaluation. In TSA-MAML, we set its

Table 2: Few-shot classification accuracy (%) of the compared approaches on the CIFARFS dataset. The reported accuracies are averaged over 600 test episodes with 95% confidence intervals.

method	1-shot 5-way	5-shot 5-way	1-shot 10-way	5-shot 10-way
Matching Net [Vinyals et al., 2016]	36.64 ± 1.13	42.68 ± 0.96	15.02 ± 1.05	32.53 ± 0.93
Meta-LSTM [Ravi and Larochelle, 2017]	41.93 ± 1.20	61.40 ± 1.15	31.40 ± 0.75	41.25 ± 0.66
Reptile [Nichol and Schulman, 2018]	51.26 ± 0.99	68.62 ± 0.98	35.73 ± 0.94	54.35 ± 0.91
HSML [Yao et al., 2019]	46.72 ± 0.87	68.76 ± 0.76	33.89 ± 0.55	53.94 ± 0.49
MMAML [Vuorio et al., 2019]	40.64 ± 0.50	49.64 ± 0.49	23.80 ± 0.28	37.19 ± 0.27
iMAML [Rajeswaran et al., 2019]	49.72 ± 0.39	60.83 ± 0.49	30.02 ± 0.36	47.13 ± 0.29
FOMAML [Finn et al., 2017]	47.03 ± 1.47	64.20 ± 1.38	34.65 ± 1.09	51.35 ± 1.16
MAML [Finn et al., 2017]	51.98 ± 0.87	68.91 ± 0.74	38.48 ± 0.55	55.24 ± 0.54
TSA-MAML	**53.07** ± **0.85**	**71.37** ± **0.74**	**39.77** ± **0.53**	**58.05** ± **0.56**
Reptile + Transduction [Nichol and Schulman, 2018]	54.03 ± 0.92	72.60 ± 0.83	38.41 ± 0.97	57.16 ± 0.87
HSML + Transduction [Yao et al., 2019]	54.71 ± 1.50	69.62 ± 1.01	38.49 ± 1.22	55.51 ± 0.68
MMAML+ Transduction [Vuorio et al., 2019]	45.16 ± 0.58	58.56 ± 0.51	27.30 ± 0.25	41.26 ± 0.26
iMAML + Transduction [Rajeswaran et al., 2019]	55.13 ± 0.38	64.44 ± 0.58	34.23 ± 0.33	49.76 ± 0.27
FOMAML + Transduction [Finn et al., 2017]	49.30 ± 1.18	66.96 ± 1.27	37.83 ± 1.06	53.23 ± 1.12
MAML + Transduction [Finn et al., 2017]	57.46 ± 0.90	72.75 ± 0.71	39.97 ± 0.56	56.21 ± 0.55
TSA-MAML + Transduction	**58.21** ± **0.93**	**73.52** ± **0.72**	**42.18** ± **0.58**	**58.69** ± **0.56**

initialization number m as three and the task number as $n = 10,000$ for clustering in k-means. For training, we use Adam [Kingma and Ba, 2014] with learning rate 10^{-3} and total iteration number $S = 40,000$. To be more stable, we use cosine annealing in [Loshchilov and Hutter, 2017] to gradually decrease the learning rate. We evaluate 600 test tasks from each sub-dataset, and test all methods on the 5-shot 5-way learning tasks under the transduction setting where test tasks share information via batch normalization, since the baselines are reported under this setting [Finn et al., 2017, Yao et al., 2019].

Results. Table 1 shows that TSA-MAML achieves the best performance over other state-of-the-arts. Specifically, on the first group-structured dataset (Aircraft + Birds + Fungi), TSA-MAML respectively makes about 2.95%, 2.84% and 3.23% improvements on the three sub-dataset (from left to right). It also brings about 4.00% improvement for the overall accuracy. Similarly, for the second group-structured dataset (Car + CUB Birds + Fungi), TSA-MAML also outperforms others on all three sub-datasets and averagely improves by about 2.36%. Compared with the approaches learning one common initialization, *e.g.* MAML and Reptile, TSA-MAML leverages task similarity in the model parameter space to discover the group structures in the tasks and learns group-specific initializations to facilitate the learning of new tasks. As a result, as shown in Fig. 1, the optimal parameters of tasks in the same group would be much closer to the group-specific initialization learnt by TSA-MAML than the single common initialization learnt for all tasks by other approaches, *e.g.* MAML. So for few-shot learning, TSA-MAML can adapt to the new tasks more quickly and better than other methods with a single initialization which confirms our theories in Sec. 3.2. Please refer to the reasons for the superiorty of our TSA-MAML over HSML and MMAML at the end of Sec. 4.2.

Indeed, we also test MAML using larger models (MAML-L). We increase its network depth from four to seven and then increase channels per layer so that new model is about 3× larger than TSA-MAML. The accuracies of MAML-L are 72.68 (aircraft), 69.73 (bird) and 54.18 (fungi) on the first group-structured dataset. In Table 5 of Appendix A, we also test MAML-L and TSA-MAML on CIFARFS [Bertinetto et al., 2019] and observe that TSA-MAML makes about at least 1.5% average improvement on the four test settings (n-way k-shot, $n = 5$ or 10 and $k = 1$ or 5) over both MAML and MAML-L. These results further demonstrate the superiority of TSA-MAML over MAML.

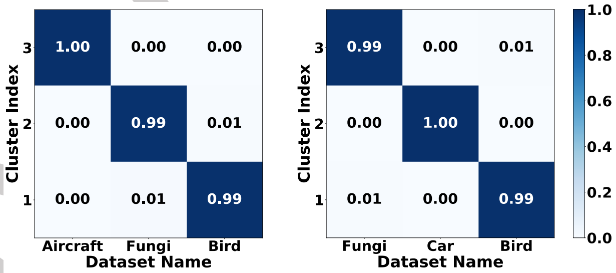


Figure 3: Usage frequency of multiple initializations in TSA-MAML on new tasks.

Fig. 3 further reports the usage frequency of the multiple initializations learnt by TSA-MAML when testing new tasks. After learning three initializations, we sample 1,000 test tasks from each sub-dataset of the group-structured dataset, and then assign one initialization for each test task by first using MAML to find its approximate optimal model θ_T and selecting a learnt initialization with smallest distance to θ_T. The values in the (i, j)-th grid in Fig. 3 denotes the frequency that TSA-MAML assigns the i-th learnt initialization to the tasks from the j-th sub-dataset. From these results in Fig. 3, one can observe that in most cases, TSA-MAML assigns the same learnt initialization for the tasks from the

Table 3: Few-shot classification accuracy (%) of the compared approaches on the tieredImageNet dataset. The reported accuracies are averaged over 600 test episodes with 95% confidence intervals.

method	1-shot 5-way	5-shot 5-way	1-shot 10-way	5-shot 10-way
Matching Net [Vinyals et al., 2016]	34.95 ± 0.89	43.95 ± 0.85	22.46 ± 0.34	31.19 ± 0.30
Meta-LSTM [Ravi and Larochelle, 2017]	33.71 ± 0.76	46.56 ± 0.79	22.09 ± 0.43	35.65 ± 0.39
Reptile [Nichol and Schulman, 2018]	**49.12 ± 0.43**	65.99 ± 0.75	31.79 ± 0.28	47.82 ± 0.30
HSML [Yao et al., 2019]	47.36 ± 0.84	66.16 ± 0.78	33.39 ± 0.57	51.53 ± 0.55
MMAML [Vuorio et al., 2019]	44.82 ± 0.46	61.47 ± 0.49	30.42 ± 0.37	48.92 ± 0.29
FOMAML [Finn et al., 2017]	48.01 ± 1.74	64.07 ± 1.72	30.31 ± 1.12	46.54 ± 1.24
MAML [Finn et al., 2017]	48.50 ± 1.83	65.93 ± 1.78	32.41 ± 1.23	48.81 ± 1.32
TSA-MAML	48.82 ± 0.88	**67.82 ± 0.72**	**34.48 ± 0.56**	**52.26 ± 0.55**
Reptile + Transduction [Nichol and Schulman, 2018]	51.06 ± 0.45	66.30 ± 0.78	33.79 ± 0.29	51.27 ± 0.31
HSML + Transduction [Yao et al., 2019]	48.82 ± 0.86	66.74 ± 0.76	34.63 ± 0.55	51.47 ± 0.54
MMAML + Transduction [Vuorio et al., 2019]	48.52 ± 0.47	64.39 ± 0.47	33.69 ± 0.35	50.90 ± 0.29
FOMAML + Transduction [Finn et al., 2017]	50.12 ± 1.82	67.43 ± 1.80	31.53 ± 1.08	49.99 ± 1.36
MAML + Transduction [Finn et al., 2017]	50.48 ± 1.81	68.06 ± 1.75	34.25 ± 1.19	51.69 ± 1.33
TSA-MAML + Transduction	**52.03 ± 0.86**	**68.97 ± 0.74**	**35.78 ± 0.58**	**52.50 ± 0.56**

same sub-dataset. This well demonstrates that TSA-MAML has leveraged the task similarity and thus can well learn the group structures in the tasks, explaining the superiority over state-of-the-arts.

5.2 EVALUATION ON THE REAL DATA

Datasets. We evaluate TSA-MAML on three benchmarks, CIFARFS [Bertinetto et al., 2019], tieredImageNet [Ren et al., 2018] and miniImageNet [Ravi and Larochelle, 2017] . CIFARFS is a recently proposed few-shot classification benchmark. It splits the 100 classes from CIFAR-100 [Krizhevsky and Hinton, 2009] into 64, 16 and 20 classes for training, validation, and test respectively. Each class contains 600 images of size 32 × 32 × 3. TieredImageNet contains 608 classes from ILSVRC-12 dataset [Russakovsky et al., 2015], in which each class has 600 images of size 84 × 84 × 3. Moreover, it groups classes into broader hierarchy categories corresponding to higher-level nodes in the ImageNet [Deng et al., 2009]. Specifically, there are total 34 top hierarchy categories which are further split into 20 training categories (351 classes), 6 validation categories (97 classes) and 8 test categories (160 classes). So all training classes are sufficiently distinct from the test classes, giving a more challenging learning task. MiniImageNet has the same image number and size for each class as tieredImageNet, but only contains 100 classes from ILSVRC-12 and also does not have hierarchy structures.

Experimental setting. We use the same network architecture, training strategy and task number n in Sec. 5.1. In TSA-MAML, the training iteration number S is $40,000$ for CIFARFS and $80,000$ for tieredImageNet and miniImageNet, and the cluster number m is five for all datasets. Like [Finn et al., 2017, Nichol and Schulman, 2018], we test all methods on 600 test episodes under (non-)transduction settings. In non-transduction, batch normalization statistics are collected from all training data and one test sample. See transduction setting in Sec. 5.1.

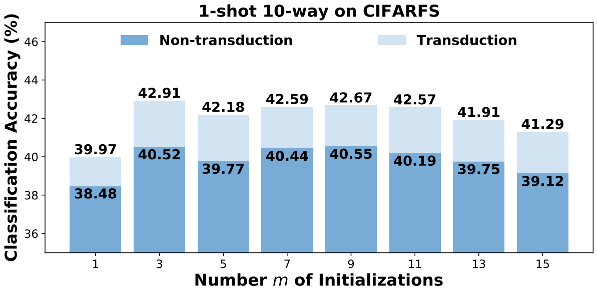


Figure 4: Effects of m to TSA-MAML.

Results. From Table 2, one can observe that TSA-MAML consistently outperforms optimization based methods, *e.g.* MAML, HSML and MMAML, and metric based method, *e.g.* Matching Net. Specifically, on *CIFARFS*, TSA-MAML respectively brings about 1.09%, 2.46%, 1.29% and 2.81% improvements on the four test cases (from left to right) under non-transduction setting, and under transduction setting it also makes about 0.75%, 0.77%, 2.21% and 2.48% improvements for the four cases. Similarly, on *tieredImageNet*, it averagely improves by about 1.68% and 1.20% on the four test cases under non-transduction and transduction cases. For the results on *miniImageNet* in Table 4 of Appendix A, TSA-MAML also respectively makes about 1.2% and 0.7% average improvement on four cases under non-transduction and transduction settings. See more details in Appendix A. These results demonstrate the advantages of TSA-MAML behind which the reasons have been discussed in Sec. 5.1. Besides, compared with MAML, TSA-MAML respectively makes about 1.73% and 1.44% average improvements on CIFARFS and tieredImageNet. These observations further confirm our theories in Sec. 3.2.

Fig. 4 shows the effects of initialization number m to the testing performance of TSA-MAML. When m ranges from 3 to 11, the performance of TSA-MAML on 1-shot 10-way learning tasks on CIFARFS are relatively stable. See similar observations of the 5-shot 10-way tasks on CIFARFS in

Fig. 5 in Appendix A. So TSA-MAML is robust to m. This is because assigning tasks into m groups means dividing model parameter space into m regions and is not hard when m is not large, as estimating approximate location of optimal task models in the parameter space is sufficient and can be achieved by MAML.

6 CONCLUSION

In this work, for the first time we theoretically justify the effectiveness of a few gradient based adaptation and the benefits of the learnt initialization for fast adaptation. Then inspired by our theory, we propose TSA-MAML as a new variant of MAML which leverages the task-similarity via learning shared initialization for similar tasks to facilitate learning new tasks. Experimental results on benchmark datasets demonstrate the superiority of TSA-MAML over the state-of-the-art methods.

ACKNOWLEDGEMENTS

Pan Zhou, Caiming Xiong and Steven Hoi are supported by the Salesforce Research. Xiao-Tong Yuan would like to acknowledge the partial support from National Key Research and Development Program of China under Grant No. 2018AAA0100400 and Natural Science Foundation of China (NSFC) under Grant No.61876090 and No.61936005.

References

R. Amit and R. Meir. Meta-learning by adjusting priors based on extended PAC-bayes theory. In *Proc. Int'l Conf. Machine Learning*, 2018.

A. Antoniou, H. Edwards, and A. Storkey. How to train your MAML. In *Int'l Conf. Learning Representations*, 2019.

Y. Bai, M. Chen, P. Zhou, T. Zhao, J. D. Lee, S. Kakade, H. Wang, and C. Xiong. How important is the train-validation split in meta-learning? In *Proc. Int'l Conf. Machine Learning*, 2021.

J. Baxter. A model of inductive bias learning. *Journal of artificial intelligence research*, 12:149–198, 2000.

Y. Bengio, S. Bengio, and J. Cloutier. Learning a synaptic learning rule. In *Int'l Joint Conf. Neural Networks* , 1990.

L. Bertinetto, J. Henriques, P. Torr, and A. Vedaldi. Meta-learning with differentiable closed-form solvers. In *Int'l Conf. Learning Representations*, 2019.

J. Deng, W. Dong, R. Socher, L. Li, K. Li, and F. Li. Imagenet: A large-scale hierarchical image database. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 248–255, 2009.

S. Du, W. Hu, S. Kakade, J. Lee, and Q. Lei. Few-shot learning via learning the representation, provably. *arXiv preprint arXiv:2002.09434*, 2020.

Y. Duan, J. Schulman, X. Chen, P. Bartlett, I. Sutskever, and P. Abbeel. RL^2: Fast reinforcement learning via slow reinforcement learning. *arXiv preprint arXiv:1611.02779*, 2016.

A. Fallah, A. Mokhtari, and A. Ozdaglar. On the convergence theory of gradient-based model-agnostic meta-learning algorithms. *arXiv preprint arXiv:1908.10400*, 2019.

C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proc. Int'l Conf. Machine Learning*, pages 1126–1135, 2017.

C. Finn, A. Rajeswaran, S. Kakade, and S. Levine. Online meta-learning. *arXiv preprint arXiv:1902.08438*, 2019.

N. Golmant. On the convergence of model-agnostic meta-learning. *http://noahgolmant.com/writings/maml.pdf*, 2019.

A. Graves, G. Wayne, and I. Danihelka. Neural turing machines. *arXiv preprint arXiv:1410.5401*, 2014.

K. Ji, J. Yang, and Y. Liang. Multi-step model-agnostic meta-learning: Convergence and improved algorithms. *arXiv preprint arXiv:2002.07836*, 2020.

Z. Kang, K. Grauman, and F. Sha. Learning with whom to share in multi-task feature learning. In *Proc. Int'l Conf. Machine Learning*, volume 2, page 4, 2011.

M. Khodak, M. Balcan, and A. Talwalkar. Provable guarantees for gradient-based meta-learning. *arXiv preprint arXiv:1902.10644*, 2019.

D. Kingma and J. Ba. Adam: A method for stochastic optimization. *Int'l Conf. Learning Representations*, 2014.

G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2, 2015.

J. Krause, M. Stark, J. Deng, and F. Li. 3D object representations for fine-grained categorization. In *Int'l IEEE Workshop on 3D Representation and Recognition*, 2013.

A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

A. Kumar and H. Daume. Learning task grouping and overlap in multi-task learning. In *Proc. Int'l Conf. Machine Learning*, 2013.

Z. Li, F. Zhou, F. Chen, and H. Li. Meta-SGD: Learning to learn quickly for few-shot learning. In *Proc. Conf. Neural Information Processing Systems*, 2017.

S. Lin, P. Zhou, X. Liang, J. Tang, R. Zhao, Z. Chen, and L. Lin. Graph-evolving meta-learning for low-resource medical dialogue generation. In *AAAI Conf. Artificial Intelligence*, 2021.

I. Loshchilov and F. Hutter. SGDR: Stochastic gradient descent with warm restarts. In *Int'l Conf. Learning Representations*, 2017.

J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Symposium on Math, Statistics, and Probability*, page 281–297, 1967.

S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi. 2018 FGCVx fungi classification challenge. *fungi-challenge-fgvc-2018*, 2018.

A. Maurer. Algorithmic stability and meta-learning. *J. of Machine Learning Research*, 6(Jun):967–994, 2005.

K. Mikhail, B. Maria-Florina, and T. Ameet. Adaptive gradient-based meta-learning methods. In *Proc. Conf. Neural Information Processing Systems*, 2019.

N. Mishra, M. Rohaninejad, X. Chen, and P. Abbeel. Meta-learning with temporal convolutions. *arXiv preprint arXiv:1707.03141*, 2(7), 2017.

T. Munkhdalai and H. Yu. Meta networks. In *Proc. Int'l Conf. Machine Learning*, pages 2554–2563, 2017.

D. Naik and R. Mammone. Meta-neural networks that learn by learning. In *Int'l Joint Conf. Neural Networks* , pages 437–442, 1992.

A. Nichol and J. Schulman. Reptile: a scalable meta-learning algorithm. *arXiv preprint arXiv:1803.02999*, 2, 2018.

A. Pentina, V. Sharmanska, and C. Lampert. Curriculum learning of multiple tasks. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 5492–5500, 2015.

A. Rajeswaran, C. Finn, S. Kakade, and S. Levine. Meta-learning with implicit gradients. In *Proc. Conf. Neural Information Processing Systems*, pages 113–124, 2019.

S. Ravi and H. Larochelle. Optimization as a model for few-shot learning. In *Int'l Conf. Learning Representations*, 2017.

M. Ren, E. Triantafillou, S. Ravi, J. Snell, K. Swersky, J. Tenenbaum, H. Larochelle, and R. Zemel. Meta-learning for semi-supervised few-shot classification. *arXiv preprint arXiv:1803.00676*, 2018.

O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, and M. Bernstein. Imagenet large scale visual recognition challenge. *Int'l. J. Computer Vision*, 115(3):211–252, 2015.

A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap. Meta-learning with memory-augmented neural networks. In *Proc. Int'l Conf. Machine Learning*, pages 1842–1850, 2016.

N. Saunshi, Y. Zhang, M. Khodak, and S. Arora. A sample complexity separation between non-convex and convex meta-learning. *arXiv preprint arXiv:2002.11172*, 2020.

J. Schmidhuber. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta hook*. PhD thesis, Technische Universität München, 1987.

J. Snell, K. Swersky, and R. Zemel. Prototypical networks for few-shot learning. In *Proc. Conf. Neural Information Processing Systems*, pages 4077–4087, 2017.

F. Sung, L. Zhang, T. Xiang, T. Hospedales, and Y. Yang. Learning to learn: Meta-critic networks for sample efficient learning. *arXiv preprint arXiv:1706.09529*, 2017.

F. Sung, Y. Yang, L. Zhang, T. Xiang, P. Torr, and T. Hospedales. Learning to compare: Relation network for few-shot learning. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 1199–1208, 2018.

S. Thrun and L. Pratt. *Learning to learn*. Springer Science & Business Media, 2012.

O. Vinyals, C. Blundell, T. Lillicrap, and D. Wierstra. Matching networks for one shot learning. In *Proc. Conf. Neural Information Processing Systems*, pages 3630–3638, 2016.

R. Vuorio, S. Sun, H. Hu, and J. Lim. Multimodal model-agnostic meta-learning via task-aware modulation. In *Proc. Conf. Neural Information Processing Systems*, pages 1–12, 2019.

C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

J. Weston, S. Chopra, and A. Bordes. Memory networks. *arXiv preprint arXiv:1410.3916*, 2014.

Y. Wu, P. Zhou, A. G. Wilson, E. Xing, and Z. Hu. Improving gan training with probability ratio clipping and sample reweighting. In *Proc. Conf. Neural Information Processing Systems*, 2020.

Y. Xiao, K. Gong, P. Zhou, G. Zheng, X. Liang, and L. Lin. Adversarial meta sampling for multilingual low-resource speech recognition. In *AAAI Conf. Artificial Intelligence*, 2021.

H. Yao, Y. Wei, J. Huang, and Z. Li. Hierarchically structured meta-learning. In *Proc. Int'l Conf. Machine Learning*, 2019.

P. Zhou and J. Feng. Understanding generalization and optimization performance of deep cnns. In *Proc. Int'l Conf. Machine Learning*, 2018a.

P. Zhou and J. Feng. Empirical risk landscape analysis for understanding deep neural networks. In *Int'l Conf. Learning Representations*, 2018b.

P. Zhou, X. Yuan, H. Xu, S. Yan, and J. Feng. Efficient meta learning via minibatch proximal update. In *Proc. Conf. Neural Information Processing Systems*, 2019.

P. Zhou, J. Feng, C. Ma, C. Xiong, S. Hoi, and W. E. Towards theoretically understanding why sgd generalizes better than adam in deep learning. In *Proc. Conf. Neural Information Processing Systems*, 2020a.

P. Zhou, C. Xiong, R. Socher, and S. Hoi. Theory-inspired path-regularized differential network architecture search. In *Proc. Conf. Neural Information Processing Systems*, 2020b.