Stress Testing of Meta-learning Approaches for Few-shot Learning

Aroof Aimen[*]   2018csz0001@iitrpr.ac.in
Sahil Sidheekh[*]   2017csb1104@iitrpr.ac.in
Vineet Madan[*]   2017csb1119@iitrpr.ac.in
Narayanan C. Krishnan   ckn@iitrpr.ac.in
Indian Institute of Technology, Ropar

Editors: Isabelle Guyon, Jan N. van Rijn, Sébastien Treguer, Joaquin Vanschoren

Abstract

Meta-learning (ML) has emerged as a promising learning method under resource constraints such as few-shot learning. ML approaches typically propose a methodology to learn generalizable models. In this work-in-progress paper, we put the recent ML approaches to a stress test to discover their limitations. Precisely, we measure the performance of ML approaches for few-shot learning against increasing task complexity. Our results show a quick degradation in the performance of initialization strategies for ML (MAML, TAML, and MetaSGD), while surprisingly, approaches that use an optimization strategy (MetaLSTM) perform significantly better. We further demonstrate the effectiveness of an optimization strategy for ML (MetaLSTM++) trained in a MAML manner over a pure optimization strategy. Our experiments also show that the optimization strategies for ML achieve higher transferability from simple to complex tasks.

1. Introduction

Rapidly learning new skills from limited experience is a fundamental trait of human intelligence. Replicating similar capabilities in deep neural networks is a challenging task. Meta-learning (ML) approaches have emerged as a promising direction in this context, facilitating learning from a limited amount of labeled training data (also referred to as the few-shot learning). They can be broadly classified into initialization and optimization strategies. Initialization strategies such as Model Agnostic Meta-learning (MAML) (Finn et al., 2017), Meta Stochastic Gradient Descent (MetaSGD) (Li et al., 2017), Task Agnostic Meta-learning (TAML) (Jamal and Qi, 2019) learn an optimal model initialization that swiftly adapts to new tasks with limited training data. Optimization strategies such as Learn2Learn (Andrychowicz et al., 2016), MetaLSTM (Ravi and Larochelle, 2017) learn parametric optimizers that accelerate the adaptation of a model to new tasks. Though the two approaches have the common objective of enforcing generalization to unseen tasks, the difference in their methodology presents a diverse set of merits and caveats.

Initialization methods learn an optimal prior on the model parameters through the experience gained across various tasks. The optimal prior is defined to be equally close to the individual training tasks' optimal parameters. This helps the model to quickly adapt to unseen tasks from the same distribution. MAML learns the optimal prior by assum-

---

[*] Equal Contribution

ing that a model quickly learns unseen tasks with sparse data if it is trained and tested under similar circumstances (Vinyals et al., 2016). Literature suggests that the optimal prior learned by MAML may still be biased towards some tasks (Jamal and Qi, 2019). TAML overcomes the bias by explicitly minimizing the inequality in the optimally initialized model performance across a batch of tasks.

However, as the task complexity increases, finding an optimal prior becomes challenging. Thus, though gradient descent takes off from a good initialization (better than random), attaining good performance also depends on the model's ability to traverse the loss surface. A good initialization alone is insufficient. Our experiments confirm the degradation in the performance of these models as the complexity of the tasks increases.

The optimization strategy MetaLSTM (Ravi and Larochelle, 2017), learns recurrent parametric optimizer that captures both task-specific and agnostic knowledge. The learned optimizer mimics the gradient-based optimization of the base model. It uses the current base model's loss and gradients to output the model parameters for the next iteration during the adaptation process. Thus, the optimizer can be viewed as employing dynamic learning rates dependent on the model's parameters and task data, unlike MetaSGD that only considers the former dependence. Optimization strategies, however, have an overhead of learning additional parameters and have limited scalability (Finn et al., 2017; Li et al., 2017). Coordinate-wise sharing of the optimizer's parameters across the base model's parameters (Andrychowicz et al., 2016; Ravi and Larochelle, 2017) reduces the learning overhead.

Initialization approaches have achieved promising results on sparse data ML settings. However, their capability on increasingly complex tasks has not been studied. In this work-in-progress paper, we conduct a stress test on initialization and optimization ML strategies against increasing task complexity. We also combine the two approaches to learn a parametric meta-optimizer MetaL-STM++, a version of MetaLSTM trained in the MAML manner. We show that MetaL-STM++ achieves significantly better performance with fewer adaptation steps on simple and complex tasks. Further, motivated by human learning tactics - experience gained from simple tasks helps to learn challenging tasks gradually- we also examine the transferability of these strategies from simple to complex tasks and vice versa.

2. Methodology

Problem Formulation and Notations

Given a principal dataset $\mathcal{D}$ and associated distribution of tasks $P(\mathcal{T})$, ML techniques create meta-sets - $\mathcal{D}^{meta-train}$, $\mathcal{D}^{meta-validation}$ and $\mathcal{D}^{meta-test}$ for training the model, tuning hyperparameters and evaluating performance. Each meta-set is a collection of mutually exclusive tasks drawn from the distribution $P(\mathcal{T})$ and each task $\mathcal{T}_i$ is associated with a dataset $\mathcal{D}_i$ comprised of disjoint sets $\{\mathcal{D}_i^{tr}, \mathcal{D}_i^{test}\}$. Each task is an N-way K-shot learning problem. ML techniques aim to learn an accurate base model f parameterized by θ for an unseen task $\mathcal{T}_i$ when fine-tuned with few examples $\mathcal{D}_i^{tr}$ of the task.

MAML uses a nested iterative process to learn the task-agnostic optimal initialization θ^*. In the inner iterations representing the task adaptation steps, θ^* is separately fine-tuned for each meta-training task $\mathcal{T}_i$ using $\mathcal{D}_i^{tr}$ to obtain θ_i through gradient descent on the loss $\mathcal{L}_i^{tr}$. Specifically, θ_i is initialized as θ^* and updated using $\theta_i \leftarrow \theta_i - \alpha \nabla_{\theta_i} \mathcal{L}_i^{tr}(f_{\theta_i})$. In the outer loop, the meta optimization is performed over θ^* using the loss $\mathcal{L}_i^{test}$ computed with the task adapted model param-

Figure 1: Oscillatory behaviour of MetaL-STM (A) compared to MetaL-STM++ (B).

eters θ_i on held-out dataset $\mathcal{D}_i^{test}$. Specifically, during meta-optimization $\theta^* \leftarrow \theta^* - \beta \nabla_{\theta^*} \sum_{\mathcal{T}_i \sim P(\mathcal{T})} \mathcal{L}_i^{test}(f_{\theta_i})$.

MetaSGD improves upon MAML by learning parameter specific learning rates α in addition to the optimal initialization in a nested iterative procedure. Meta-optimization is performed on θ^* and α is learned in the outer loop using the loss $\mathcal{L}_i^{test}$ computed on held-out dataset $\mathcal{D}_i^{test}$. Specifically, during meta-optimization $(\theta^*, \alpha) \leftarrow (\theta^*, \alpha) - \beta \nabla_{(\theta^*, \alpha)} \sum_{\mathcal{T}_i \sim P(\mathcal{T})} \mathcal{L}_i^{test}(f_{\theta_i})$. Learning dynamic learning rates for each parameter of a model makes MetaSGD faster and more generalizable than MAML. A single adaptation step is sufficient to adjust the model towards a new task. However the dependence of the learning rate only on model parameters limits the capability of MetaSGD.

TAML reduces the bias of the optimal initialization, learned through MAML, towards any task by explicitly minimizing the inequality among the performances of a batch's tasks. It uses statistical measures like the Theil index, Atkinson index, Generalized entropy index, and Gini coefficient to estimate the inequality among tasks' performances. In this work, we use the Theil index owing to the average best results, defined as $\frac{1}{B} \sum_{i=1}^{B} \frac{\mathcal{L}_i^{test}}{\bar{\mathcal{L}}^{test}} \ln \frac{\mathcal{L}_i^{test}}{\bar{\mathcal{L}}^{test}}$ to measure the in-

equality among the performances of the tasks in a batch. B is number of tasks in a batch, $\mathcal{L}_i^{test}$ is loss of task $\mathcal{T}_i$ on $\mathcal{D}_i^{test}$ and $\bar{\mathcal{L}}^{test}$ is the average test loss of a task batch. For few-shot learning, TAML proposes entropy minimization to preclude the bias of θ^* towards any task $\mathcal{T}_i$ by maximizing the entropy of the labels predicted by f_{θ^*} and minimizing the entropy of the labels predicted by adapted model f_{θ_i}. This is equivalent to a maximum entropy prior over θ^* such that the initialized model is not biased to any task.

MetaLSTM learns an optimizer $\mathcal{M}$ parametrized by ϕ to supervise the optimization process of the base model (f_θ). The parametric optimizer is an LSTM, which is inherently capable of performing bi-level learning due to its architecture. During adaptation of f_θ on $\mathcal{D}_i^{tr}$, $\mathcal{M}$ takes meta information characterized by current loss $\mathcal{L}_i^{tr}$ and gradients $\nabla_{\theta_i}(\mathcal{L}_i^{tr})$ as input and outputs the next set of parameters θ_i. Internally, the cell state of $\mathcal{M}$ corresponds to θ, and the cell state update in $\mathcal{M}$ resembles a learned and controlled gradient update as the emphasis on the previous parameters and the current update is regulated by the learned forget and input gates respectively. While adapting f_θ to $\mathcal{D}_i^{tr}$, meta-information about the trajectory on the loss surface across the adaptation steps is captured in the hidden states of $\mathcal{M}$, representing the task-specific information. During meta-optimization, ϕ is updated based on loss $\mathcal{L}_i^{test}$ of task computed on held-out dataset $\mathcal{D}_i^{test}$ to garner the common knowledge across tasks.

MetaLSTM++ A caveat of MetaL-STM is the sequential update to $\mathcal{M}$ after each adaptation task. As a result, the optimization strategy traverses the loss surface in an ordered sequence of task specific optima. This leads to a longer and oscillatory optimization trajectory as shown in Figure 1 and bias of $\mathcal{M}$ towards the final task. We propose to overcome this bottleneck by learning

40

	Test Accuracy (20-Way)	
Model	1 Shot	5-Shot
MAML*	91.93 ± 0.72	97.65 ± 0.20
MetaSGD*	94.58 ± 0.59	97.79 ± 0.23
TAML (Theil Index)*	92.25 ± 0.70	95.14 ± 0.87
MetaLSTM*	90.63 ± 0.83	97.11 ± 0.24
MetaLSTM++ (Ours)	**96.50 ± 0.42**	**98.41 ± 0.31**

Table 1: Few shot classification performance of ML algorithms on the Omniglot dataset. The ± represents the 95% confidence interval across 300 tasks. All the algorithms are rerun (denoted by *) on the same split for a fair comparison.

$\mathcal{M}$ according to the training procedure of initialization ML strategies, termed as MetaLSTM++. Unlike the MetaLSTM, the $\mathcal{M}$ of MetaLSTM++ is updated based on the average test loss of a task batch. This is intuitive as a batch of tasks may better approximate the data distribution, instead of a single task. The batch update on $\mathcal{M}$ makes the optimization trajectory smooth, short, and robust to task order (Figure 1).

3. Experiments and Results

The ML approaches are benchmarked using the popular Omniglot dataset (Lake et al., 2015). We consider this dataset owing to its simplicity and show that the performance of ML strategies on complex settings has not saturated even on this dataset. We follow the standard split (1200 : 423) of the dataset keeping 220 classes from the meta-training split to tune the models' hyperparameters. The images are downsampled to 28 ×28. We use the same architecture as in (Finn et al., 2017) for the base model. We use a two-layer LSTM following Ravi and Larochelle (2017) for the parametric optimizer.

The hyper-parameters for each ML approach have been fine-tuned for each complexity setting separately for a fair comparison. We find optimal hyper-parameters by performing a grid search over 30 different configurations for 5000 iterations. The search interval for all strategies is the same. The number of adaptation steps range within $\left[2^2 - 2^6\right]$, the meta and base learning rates follow log uniform distribution in the ranges $\left[1e^{-4} - 1\right]$ and $\left[1e^{-4} - 1e^{-2}\right]$ respectively. For TAML (Theil), λ also follows a log uniform distribution over the range of $\left[1e^{-2} - 1\right]$. The meta batch size was varied among 4, 8, 16, and 32. The final model was trained for 20000 iterations using the optimal set of hyper-parameters. Early stopping was employed if no improvement was observed for 2500 steps. Cross-entropy loss is used in all models.

We verified our implementation of all the ML approaches on 20 way 1 and 5 shot settings. The results reported in Table 1 are consistent with the findings in the existing literature. It is evident from the results that MetaLSTM++ outperforms the state of the art ML approaches in both 1 and 5 shot settings, thus showing the promise of combining the initialization and the optimization strategies for ML.

3.1. Increasing Task Complexity

We use the most challenging few-shot learning setting - one-shot setting, for the next set of experiments. Furthermore, increasingly challenging learning tasks are designed by increasing the number of classes. In particular, we consider 20, 40, 60, 80, 100, 150, 175, and 200 classes. This challenging setting was never studied before. Figure 2 shows the results for MAML, TAML, MetaSGD, MetaLSTM, and MetaLSTM++ as the number of ways in a task increases. The most salient observation is the rapid decrease in
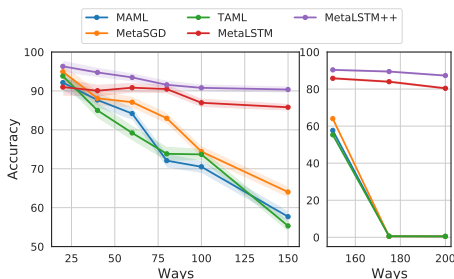
Figure 2: Accuracy of ML strategies w.r.t the increase in the number of ways in a task. The confidence interval is 99.9%.
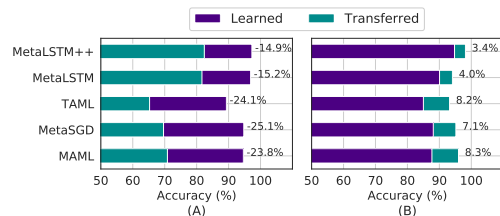

Figure 3: Transferability of ML approaches from (a) simple to complex tasks (40-way 5-shot to 40-way 1-shot) (b) complex to simple tasks (40-way 1-shot to 40-way 5-shot)

MAML, TAML, and MetaSGD performances against increasingly challenging tasks. Further, an accelerated fall in their performance is witnessed when the number of ways in a task rises beyond 150. Despite the exhaustive hyperparameter search and finetuning, the initialization based approaches saturate at accuracies close to zero for 175 and 200 way tasks. This indicates that an optimal initialization alone is insufficient for the model to perform well on complex tasks. A justification for this behavior is that the optimal parameters for the diverse tasks may lie far from each other, so finding an initialization that lies in the proximity of all tasks may be difficult. Among initialization based methods, MetaSGD is comparatively robust to the increasingly complex tasks. A possible explanation for this could be that MetaSGD learns parameter adaptive learning rates in addition to the optimal initialization, which helps the model traverse the loss surface to some extent while meta-testing.

MetaLSTM and MetaLSTM++, on the contrary, exhibit a marginal decline in performance despite the increasing task complexity. This indicates that the parametric optimizers can learn the loss surface's generic dynamics to assist the model to approach the optima, even for complex tasks.

It is evident that MetaLSTM++ consistently outperforms all ML approaches, indicating that the parametric optimizer trained according to the initialization strategy has the edge over pure optimization and initialization strategies.

3.2. Transferability across Complexity

The primary goal of ML approaches is to learn from experience a prior that generalizes well. We investigate the generalizability of ML approaches on complex tasks, using prior learned from simpler tasks. Specifically, we meta-train a model on a 40-way 5-shot setting and meta-test it on a 40-way 1-shot setting. A 5-shot learning task is less challenging than a 1-shot task. We observe a performance drop across all ML strategies as expected, as illustrated in Figure 3(A). However, the critical observation is that initialization strategies - MAML, MetaSGD, and TAML experience a substantial performance reduction, indicating that optimal initialization obtained for a model in a simple setting is not adequately generalizable to a complex setting. However, MetaLSTM and MetaLSTM++ show lesser performance reduction, confirming the generalizability of a parametric optimizer from simple to complex

settings. MetaLSTM++ consistently outperforms other approaches.

We also investigate the reverse transferability of ML models from complex to simpler tasks. In particular, we train and test a meta-model on a 40-way 1-shot and 5-shot tasks respectively. As the model sees more information during the meta-test time, we expect all the models' performance to increase. The results from Figure 3(B) show that the increase in the performance of MAML, TAML, and MetaSGD is higher than MetaLSTM and MetaLSTM++. However, MetaLSTM++ still achieves higher overall accuracy across both scenarios.

To rule out the inadequacy of the adaptation steps during meta-testing, we observe the behavior of 40-way 1-shot trained models against an increasing number of adaptation steps during meta-testing on 40-way 5 shot tasks. The test accuracies are averaged across 300 tasks. We observe from Figure 4 (a) that initialization methods and MetaLSTM++ require less adaptation on the test data to achieve peak performance, but MetaLSTM requires significantly more adaptation steps. We also notice that MetaLSTM++ performs better than all ML strategies throughout the scenario (achieves higher accuracy in a lesser number of adaptation steps).

3.3. Ablation on the Initialization

MetaLSTM and MetaLSTM++ implicitly learn a good initialization for the base learner. We study the effect of decoupling the learned initialization from the parametric optimizer in a 20-way 5-shot setting. In the first experiment, we switch the MetaLSTM and MetaLSTM++ optimizers with Adam while retaining their initializations. This setting is comparable to MAML. We observe from Figure 4 (b) that a model initialized with MetaLSTM cell state and adapted
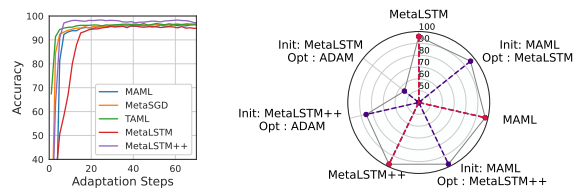

Figure 4: Ablation studies across: (a) Adaptation steps (b) Initialization (radius = test accuracy across 300 tasks)

with Adam performs significantly poor at the test time, indicating that the initialization learned by MetaLSTM is substandard. We also observe that MetaLSTM++ learns a better initialization than MetaLSTM owing to its meta-training strategy; however, MAML learns the most desirable initialization. In the second experiment, we investigate the model's performance when initialized by MAML and guided by MetaLSTM and MetaLSTM++ optimizers. The results from Figure 4 (b) show no substantial increase in the performance, indicating that the learned parametric optimizers can guide the model to superior performance even with substandard initialization (MetaLSTM and MetaLSTM++).

4. Conclusion

Meta-learning approaches for few-shot learning have shown promising results on simpler tasks of the Omniglot benchmarking dataset. In this paper, we conduct a stress test on these approaches using more challenging one-shot learning tasks from the same dataset. We observe a sharp drop in the accuracy (close to 0!) for MAML, TAML, and MetaSGD with a significant increase in the number of classes (175). Surprisingly, optimization strategies like MetaLSTM and the proposed variant MetaLSTM++ continue to

maintain accuracies above 80% even with the challenging task of 200-way 1-shot learning. The experiments on transferability of meta-models from simpler to complex tasks (and vice versa) also suggest the effectiveness of optimization strategies over initialization. While the results show the limitations of and motivate future research on pure initialization strategies for meta-learning, it also warrants the study into the causes for optimization strategies' effectiveness.

Acknowledgments

References

Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. Learning to learn by gradient descent by gradient descent. In *NIPS*, 2016.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017.

Muhammad Abdullah Jamal and Guo-Jun Qi. Task agnostic meta-learning for few-shot learning. In *CVPR*, 2019.

Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 2015.

Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-sgd: Learning to learn quickly for few-shot learning, 2017.

Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *NIPS*, 2016.