学生信息：吕行健，上海星河湾双语学校，中国

导师信息：张伟楠，上海交通大学，中国

报告标题：Meta-Learning Algorithms for Multi-task Data Generation

Meta-Learning Algorithms for Multi-task Data Generation

Xingjian Lyu
Shanghai Starriver Bilingual School
lyuxingjian@qq.com

Abstract

Generative Adversarial Networks (GAN) are able to implicitly model distributions, and demonstrate great potential on tasks such as image generation, image translation, etc. However, GAN requires substantial data and displays slow convergence during training, making the training process time-consuming as well as data-consuming. On the other hand, recent advances in meta-learning algorithms make fast adaptation possible for supervised-learning tasks. In this work, we consider combining GAN with various meta-learning algorithms to enable data and time-efficient GAN adaptation. Few-shot image generation experiments on MNIST and Omniglot show for the first time that Model Agnostic Meta-Learning (MAML) can be incorporated with the GAN framework to yield competitive results. We also investigate the possibility of combining GAN with First-Order MAML and Meta-SGD, and point out appropriate practices in implementation for each. Finally, we propose using a pretrained classifier as a critic to quantitatively evaluate meta-trained GANs' performance on few-shot generation tasks, and report state-of-the-art few-shot generation performance.

1 Introduction

Deep Learning methods have achieved human-level performance on various tasks [8, 16]; specifically, Generative Adversarial Networks (GAN) are able to implicitly model distributions and excel at generating data such as images [8]. While state-of-the-art GANs generate samples indistinguishable by humans, their training processes are extremely data-consuming and time-consuming [8, 16]. Despite GANs' compelling ability to synthesize data, there is often little value in doing so if the data size that enables effective GAN training is readily available. In this respect, GAN has not yet fully bridged the gap between artificial and human intelligence. The latter is capable of leveraging past experience for much faster learnin. For example, a middle-schooler seeing the character φ for the first time can quickly generalize to drawing it through a few samples, while GANs require more training time and substantially more training instances to produce results of similar quality.

We observe that this disrepancy in generalization power results from the fact that humans utilize prior information obtained from past seen tasks, while GAN is trained with no prior knowledge: the middle-schooler has drawn hundreds of distinct characters before seeing φ, while GAN is trained from randomly initialized weights. This explanation suggests that to achieve human-level adaptation performance, we must somehow allow GAN to utilize prior knowledge. Specifically, GAN should leverage prior knowledge obtained from tasks seen before to improve its generalization performance over similar unseen tasks. Fortunately, meta-learning, or learning to learn [27], provides a readily studied framework under which past experiences can be integrated with novel information: meta-learning algorithms extract transferrable knowledge from a multitute of seen tasks to optimize performance on novel tasks [26, 28], and have achieved state-of-the-art performance in many supervised and reinforcement learning problems [4]. Some meta-learning algorithms learn update functions [3, 7, 26], while recent advances show the potential of meta-learning algorithms that learn model initializations [10, 20]; certain works show how both can be learned at the same time [4, 18]. Initialization-based meta-learning algorithms have been receiving increasing attention; not only do they achieve impressive performance in supervised learning scenarios e.g. few-shot image classification [4] and reinforcement learning, they also provide insight on combining powerful models with prior information to allow fast adaptation [10]. Therefore, it is compelling to combine meta-learning algorithms with the GAN training framework. In this work, we limit our investigation to initialization-based meta-learning algorithms, specifically MAML, First-Order MAML (FOMAML), and Meta-SGD, because of their simplicity and good performance [4, 10, 18, 20].

However, there still exist practical obstacles in incorporating such algorithms with GAN, due to the nature of adversarial training and the meta-learning setup. First, the combined meta-learning GAN framework proves difficult to train. Previous works have identified significant training difficulty on the part of both meta-learning and GAN algorithms [4, 5], and our experiments corroborate that straightforward combination of several meta-learning algorithms and GAN constantly results in training failure. Second, there exists no quantitative measure for models' performance on few-shot generation tasks; human evaluations of image quality have high variance and are potentially biased.

Our work investigates and provides practices that overcome these aforementioned obstacles, and reports the first conception and successful combination of MAML with GAN to the extent to our knowledge. Our work shows that MAML can be used in a more general sense to optimize nested objectives as opposed to a single loss function. We note that Multi-Step Loss optimization (MSL), a stabilizing technique for MAML proposed by Antoniou et al., greatly stabilizes training for our purposes [4]. Besides

The 1st International Conference on Distributed Artificial Intelligence (DAI 2019), Beijing, China.

MAML, we also investigate the performance of FOMAML, and Meta-SGD when combined with GAN for image generation on MNIST and Omniglot [17]. We report results and explanation for the results of each algorithm. Our work also contributes a novel quantitative measure of using a pretrained classifier to evaluate meta-trained GANs' few-shot generation performance. The classifier was pretrained on training data for generation along with its labels, and we use the critic to calculate the mean Negative Log Likelihood (NLL) of generated images. We showcase improved performance based on this metric compared to strong baselines.

The applications of the combination of meta-learning and GAN extend beyond few-shot data generation: it is suitable for any GAN scenarios where there exists a multitude of similar tasks to train on, yet at the same time limitations such as data size or computational expenses that discourage conventional GAN implementation. Our experiments show that compared to GAN that is trained from scratch, meta-trained GAN displays much faster adaptation and significantly better generalization performance when few data samples are available.

2 Related Works

Many well-studied meta-learning algorithms learn an optimizer for task adaptation [3, 7, 26]. Our work is mainly concerned with MAML, FOMAML, and Meta-SGD, three initialization-based meta-learning algorithms which achieve state-of-the-art generalization performance on various tasks. Finn et al. proposed MAML, a principled meta-learning algorithm that optimizes validation loss with respect to model initialization; the work called attention the potential of initialization-based approaches to meta-learning, and reported much improved performance in both few-shot classification and reinforcement learning[10]. Finn et al. also proposed FOMAML, a first-order approximation of MAML which does not require computing the Hessian while achieving comparable results on few-shot classification tasks. Antoniou et al. noted MAML's training instability, and contributed various modifications to stabilize MAML training and achieve better generalization performance [4]. Nichol et al. proposed reptile, a first-order meta-learning algorithm that also optimizes across-task generalization [20]. Li et al. proposed meta-SGD, which extends meta-learned parameters to learning rates for each parameter, and uses only one inner-loop update; Li et al. showcases significantly better performance than MAML on Omniglot and Mini-Imagenet classification[18, 23].

Few-shot generative models have also been intensively studied. Bartunov and Vetrov et al. used matching networks and variational autoencoders for few-shot image generation [6]. Gregor et al. developed Deep Recurrent Attention Writer (DRAW) [12], and Rezende et al. proposed using a sequential generative model to generalize it for few-shot learning [25]. The generation quality of DRAW is compelling, but DRAW is limited to generating images, specifically, characters. Liu et al. demonstrated realistic pet-swap results with Few-Shot Unsupervised Image-to-Image Translation (FUNIT) [19]. FUNIT uses conditional GAN with architecture engineered for image translation; the prior of translation tasks is encoded by the FUNIT architecture and training procedure, thus FUNIT is limited in its few-shot applications to image translation tasks. Clouatre et al. proposed Few-shot Image Generation with Reptile (FIGR), which combines reptile with GAN for few-shot image generation [9]; the work reported visual results, but did not quantitatively evaluate generation quality. We use reptile trained with WGAN, as proposed by Clouatre et al., as an advanced baseline against which we evaluate our model. Reed et al. first combined MAML with autoregressive models to learn the distribution over task-specific distributions, yet did not leverage GAN's capability to efficiently model distributions [24].

We did not come across any published literature that combines MAML and GAN for multi-tasking, or systematically investigates the possibility of combining initialization-based meta-learning algorithms with GAN; we also note the absense of a quantitative metric for few-shot generation that is widely applicable. Our work makes the unique contribution of formulating the combination of MAML and GAN, pointing out revisions vital to its training stability and its state-of-the-art few-shot generation performance. We also report results of combining GAN with FOMAML and meta-SGD.

3 Preliminaries

3.1 Generative Adversarial Networks

Generative Adversarial Network (GAN) has achieved impressive results on tasks including data generation and image super-resolution [1, 2]. Conventional GAN training framework consists of simultaneous training of a generator G and a discriminator D [11]. D seeks to distinguish between data generated by G and real data, while G seeks to learn a mapping from prior distribution z to the data distribution. GAN's training objective is formulated in (1), where P_data is the underlying data distribution, and x is data sampled from P_data.

$$\min_G \max_D \mathbb{E}_{x\sim P_{data}}[\log(D(x)] + \mathbb{E}_{z\sim P_z}[\log(1 - D(G(z)))] \tag{1}$$

One variant of GAN we use is Wasserstein GAN (WGAN), proposed by Arjovsky et al. [5]. We use gradient penalty enforce the Lipschitz constraint [13]. The WGAN objective is shown as follows:

$$\min_G \max_{||D||_L \leq 1} \mathbb{E}_{x\sim P_{data}}[D(x)] - \mathbb{E}_{z\sim P_z}[D(G(z))] \tag{2}$$

Arjovsky et al. proved that optimizing objective as in (2) is equivalent to minimizing the Wasserstein distance between $G(\boldsymbol{z})$ and $\boldsymbol{x}$, and that Wasserstein distance has better theoretical properties that help stabilize GAN training. Our experiments corroborate Arjovsky et al.'s claim that WGAN improves training stability: While objective (1) only accepts MAML, (2) can incorporate both MAML and reptile. We find that using the wasserstein objective results in better NLL performance, so uses it by default for our experiments.

We base our GAN architecture on DCGAN proposed by Radford et al. for our experiments becuase it offers stable training [22]; the model, as shown in figure 1, uses LeakyReLU with α=0.2 for activation, and omits normalization layers.

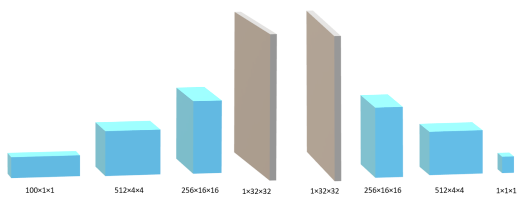


Figure 1: GAN architecture

3.2 Model-Agnostic Meta-Learning

Model-Agnostic Meta-Learning (MAML) optimizes model's task-specific loss with respect to its initialization [10]. The MAML training framework resembles those of other meta-learning algorithms we investigate in this work: in the inner-loop, the model starts from the meta-learned initialization to adapt quickly to each task in a meta-batch. The outer-loop then optimizes the validation loss computed using the task-adapted parameters with respect to the initialization. MAML's loss is shown in below.

$$L_{MAML} = \mathbb{E}_{\tau \sim P_\tau}[L_\tau(\phi_\tau^k)]$$

Where ϕ is the model initialization to be learned in the meta-learning process, and P_τ is a distribution of tasks such that each task τ has a loss function L_τ that evaluates model parameters. ϕ_τ^k is the parameter that has been updated k steps on task τ using gradient-based optimizers that can be differentiated through.

Because MAML requires computing the Hessian of the loss with respect to the parameters, it is memory and computation-demanding. Finn et al. proposed a First-Order approximation of MAML (FOMAML), and noted that it significantly reduces computational overhead while yielding results similar to MAML [10]. Nichol et al. pointed out is equivalent to ignoring the Jacobian of SGD update: $\nabla_\phi L_{MAML}$ is approximated using $\nabla_{\phi_\tau^k} L_{MAML}$, which only requires computing the gradient [20].

However, MAML suffers from training instability, and works have observed that MAML is sensitive to model architecture and hyperparameters [4]. Antoniou et al. proposed Multi-Step loss Optimization (MSL), revising the MAML loss to include not only the validation loss for the final step, but also weighted loss for every intermediate step. The MAML-MSL objective is shown in (3).

$$L_{MAML}^{weighted} = \mathbb{E}_{\tau \sim P_\tau}[\sum_{i=1}^{k} \beta_i L_\tau(\phi_\tau^i)] \tag{3}$$

where β is a weight vector for each inner-loop step. Antoniou et al. suggests annealing the MSL objective to the original MAML objective as training progresses, but we find this practice detrimental to MAML-GAN training stability. A special case of (3) is when β is uniform, i.e. equal weights are assigned to each inner-loop step; The uniform MAML-MSL objective is shown in (4).

$$L_{MAML}^{uniform} = \mathbb{E}_{\tau \sim P_\tau}[\frac{1}{k} \sum_{i=1}^{k} L_\tau(\phi_\tau^i)] \tag{4}$$

For our work, we use the uniform MAML-MSL objective, and we observe stability improvement over the both original MAML objective and annealed MAML-MSL objective.

3.3 Meta-SGD

Li et al. proposed meta-SGD, which learns the update direction as well initialization; meta-SGD limits task-specific update to a single step[18]. Li et al. argued that learning per-parameter learning rate allows greater adaptation capacity, and showcased competitive generalization performance on Omniglot and mini-imagenet. Meta-SGD's loss is shown in (5).

$$L_{meta\text{-}SGD} = \mathbb{E}_{\tau \sim P_\tau}[L_\tau(\phi - \alpha \circ L_\tau(\phi))] \tag{5}$$

where α is the learning-rate vector, and $\circ$ denotes element-wise multiplication. Meta-SGD optimizes $L_{meta-SGD}$ w.r.t. both α and ϕ; note that the task-specific update direction, while solely dependent on the gradient, is not necessarily the same as the latter's.

4 Methods

4.1 Multi-task GAN using MAML

In this section, we formulate the multi-task GAN problem and show that it naturally incorporates MAML. Recall that to model a single distribution $\boldsymbol{X}$, GAN optimizes the following objective, where G and D are parameterized by θ and ϕ respectively:

$$V(\theta, \phi) = \mathbb{E}_{x \sim \boldsymbol{X}}[D^\phi(x)] - \mathbb{E}_{z \sim P_z}[D^\phi(G^\theta(z))] \tag{6}$$

$$\min_\theta \max_{\phi, ||D_\phi||_L \leq 1} V(\theta, \phi)$$

Modifications to objective (6), i.e. applying GAN variants that have different single-task objectives, are straightforward extensions, and do not affect the conclusion.

Multi-task GAN learns a distribution over distributions. We denote each distribution τ and the meta-distribution P_τ. In the case of Omniglot, samples drawn from τ are images of a character, while τ is drawn from P_τ, distribution over characters. Thus the multi-task objective is as follows, where θ_τ and ϕ_τ are specific to each task:

$$\mathbb{E}_{\tau\sim P_\tau}[\min_{\theta_\tau}\max_{\phi_\tau} V(\theta_\tau, \phi_\tau)] \tag{7}$$

MAML constrains the task-adapted parameters such that they are updated from the meta-learned initialization. Similarly, we constrain θ_τ and ϕ_τ such that they are updated from initializations θ and τ to optimize $V(\theta_\tau, \phi_\tau)$. θ_τ and ϕ_τ can be rewritten as follows, where $\Delta\theta_\tau$ and $\Delta\phi_\tau$ optimizes $V(\theta_\tau, \phi_\tau)$:

$$\theta_\tau = \theta + \Delta\theta_\tau$$
$$\phi_\tau = \phi + \Delta\phi_\tau$$

Therefore (7) can be rewritten as:

$$\mathbb{E}_{\tau\sim P_\tau}[\min_{\theta}\max_{\phi}\min_{\Delta\theta_\tau}\max_{\Delta\phi_\tau} V(\theta_\tau, \phi_\tau)] \tag{8}$$

Because θ and ϕ are independent of τ:

$$\min_{\theta}\max_{\phi}\mathbb{E}_{\tau\sim P_\tau}[\min_{\Delta\theta_\tau}\max_{\Delta\phi_\tau} V(\theta_\tau, \phi_\tau)] \tag{9}$$

Just as we optimize the GAN mini-max objective using gradient descent, we can approximate the mini-max expression inside the expectation with k gradient descent iterations. Using U_τ^k as an operator that updates parameters for k steps on task τ, the multi-task GAN objective under MAML becomes:

$$\min_{\theta}\max_{\phi}\mathbb{E}_{\tau\sim P_\tau}[V(U_\tau^k(\theta), U_\tau^k(\phi))] \tag{10}$$

Operator U can be differentiated through by passing a gradient through a gradient, which is readily implemented in standard deep learning libraries such as Pytorch [21]. Thus we can optimize objective (10) using gradient descent. For our work, we use RMSprop as the optimizer for both inner-loop and outer-loop objectives. The MAML-GAN algorithm is shown in 1.

4.2 Evaluating meta-trained GANs

We propose using a pretrained classifier as a critic to estimate the Negative Log-Likelihood (NLL) of the generated images, and using classification accuracy as an intuitive metric of generation quality. Averaging NLL over unseen test tasks as shown in (11) provides a quantitative metric to evaluate the performance of meta-trained GANs.

$$NLL(G) = \mathbb{E}_{\tau\sim P_\tau, x\sim G(z)}[-\log(P(\tau|x))] \tag{11}$$

G is the generator adapted on τ, and $P(\tau|x)$ is given by the critic. We note that the critic assigns a maximum likelihood of 1 and a minimum of 0 for any x, therefore the proposed metric has a minimum of 0 and no upper bound.

We approximate the expectation in (11) by sampling 256 test tasks (with replacement), and computing the average NLL and accuracy on 4096 generated samples for each task after 10-step adaptation.

A ResNet with four residual blocks is trained as the critic [14]. The model achieves 99.4% and 81.7% accuracy on MNIST and Omniglot, respectively. Model architecture is shown in figure 2.

Algorithm 1 MAML-GAN

Require: P_τ: distribution over generation tasks
Require: γ, β: outer and inner-loop learning rates
Require: N, k: meta batch size, no. of inner-loop steps
Require: ϕ, θ: parameters of D and G
Require: $RMSprop(L, \phi, \alpha)$: Update ϕ in direction of L with learning rate α

Randomly initialize ϕ, θ
while not converge **do**
  sample $\{\tau_1, \tau_2 \ldots \tau_N\} \sim P_\tau$
  $L_D^{meta}, L_G^{meta} = 0$
  for τ_i **in** $\{\tau_1, \tau_2 \ldots \tau_N\}$ **do**
    $\phi_i, \theta_i = \phi, \theta$
    for inner-step **in** range(k) **do**
      sample $x \sim \tau_i, z \sim P_z$
      calculate D_{loss}, G_{loss} using ϕ_i, θ_i
      $\phi_i \leftarrow RMSprop(D_{loss}, \phi_i, \beta)$
      $\theta_i \leftarrow RMSprop(G_{loss}, \theta_i, \beta)$
      resample $x \sim \tau_i, z \sim P_z$
      calculate $D_{loss}^{val}, G_{loss}^{val}$ using ϕ_i, θ_i
      $L_D^{meta} \leftarrow L_D^{meta} + \frac{1}{Nk} D_{loss}^{val}$
      $L_G^{meta} \leftarrow L_G^{meta} + \frac{1}{Nk} G_{loss}^{val}$
    end for
  end for
  $\phi \leftarrow RMSprop(L_D^{meta}, \phi, \gamma)$
  $\theta \leftarrow RMSprop(L_G^{meta}, \theta, \gamma)$
end while

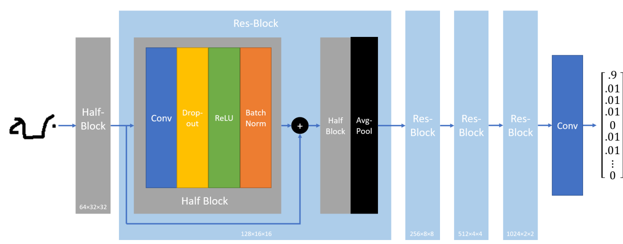


Figure 2: ResNet critic architecture

5 Experiments

We combine GAN with MAML, FOMAML, and Meta-SGD on MNIST and Omniglot [17]. For all our experiments, β and γ are set to 0.0001, and k to 10; refer to algorithm 1 for the hyperparameters. Training loss plots of the reported experiments can be found in the appendix.

Dataset MNIST consists of 60000 training and 10000 test images, split evenly across 10 digits. Due to the unsupervised nature of our task, we combine the training and test images for each digit. Digits 0 to 6 are used for meta-training, and 7 to 9 for evaluation. Thus, model has access to 7000 images for each of seven training tasks, and is required to generate images for three evaluation tasks using limited instances and gradient steps.

Omniglot consists of 1623 characters and 32460 images;

there are 20 images for each distinct character. Omniglot is a much more challenging few-shot generation problem than MNIST because not only are there more generation tasks, there are also less instances to train on per-task and for the meta-learning process in general. We comply with the train-test task split originally offered by the dataset: model is meta-trained on 964 characters, and evaluated on 659 characters. Examples of Omniglot task images are shown in figure 3. The first five columns contain images from training tasks, and the right, test tasks.

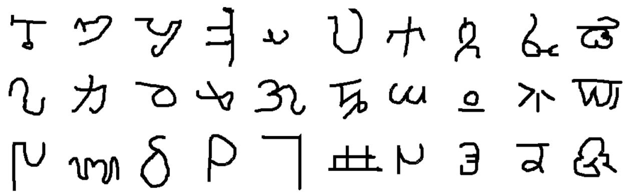


Figure 3: Omniglot task images

Failed Combinations Despite our attempts, combination of meta-SGD with GAN fails to adapt quickly to each task. First, meta-SGD cannot be trained with the wasserstein objective; we use objective (1) for the reported experiments. Second, as shown in figure 4, though the model generate reasonable images, the adapted parameters fail to generate distinct characters for each task: adapted generations of digits 7 (columns 4-6), 8 (columns 7-9), and 9 (columns 10-12) are all similar. We hypothesize that meta-SGD's training failure may be due to the disrepancy between the algorithm's adaptation capacity and the complexity of task-specific GAN training. Meta-SGD adapts to a new task in one update, which might not be enough for effective GAN adaptation.

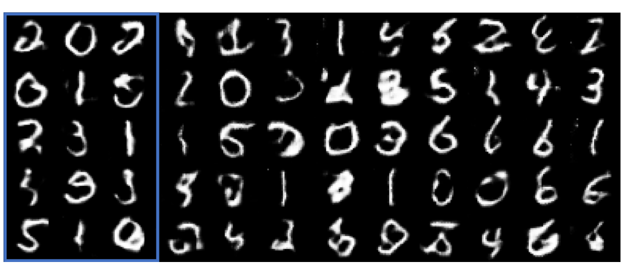


Figure 4: MNIST digits generated by metaSGD-GAN. Prior generations are boxed in blue

Successful Combinations Under algorithm 1, we are able to successfully train DCGAN with MAML; MAML-GAN yields most stable training and meaningful generated images.

While DCGAN is also capable of generating realistic digits, taken sufficient training, MAML-GAN demonstrates much greater adaptation efficiency. Figure 5 highlights the advantage offered by MAML meta-training: The upper half of the figure showcases generated images after meta-trained GAN is adapted on unseen digits; generation of meta-trained initialization is boxed in blue. The lower half is identical in setup except that it reports generation using random initialization. One can see that while randomly initialized GAN just began to trace the digits' approximate shapes, meta-trained GAN has already begun to produce discernable digits.

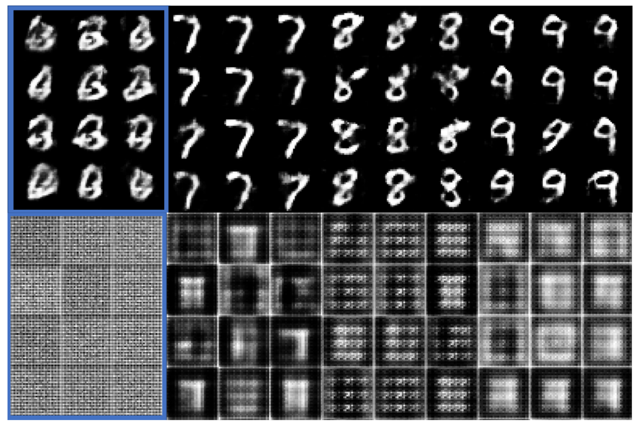


Figure 5: MNIST digits generated by MAML-GAN. Upper half contains generated images of model adapted on unseen digits for 10 steps; lower half contains generated images of randomly initialized GAN using the same setup. Prior generations are boxed in blue.

MAML-GAN also achieves impressive generalization results on Omniglot. Figure 6 demonstrates that the meta-trained model is capable of quickly adapting to to the structure of both simple as well as complex characters. We note that one important challenge Omniglot offers is the scarcity of training samples: each Omniglot character contains only 20 images. As shown in figure 6, randomly initialized DC-GAN overfits the task after being sufficiently trained (500 updates). The overfitting model generates blurry images and proves unable to capture the structure of the character.

We are also able to use FOMAML, which reduces MAML's computational burden. We note that compared to MAML, FOMAML's losses fluctuate greatly, and the meta-trained initializations generate very different images: while MAML priors resemble the characters meta-trained on, as shown in figure 5, FOMAML priors are comparatively meaningless and distinctly different in various stages of training. FOMAML training is more unstable than both MAML and reptile. Interestingly, when FOMAML-GAN converges on MNIST, it significantly outperforms both MAML and reptile; however, FOMAML's losses are unstable on Omniglot; correspondingly it underperforms them on the dataset.

5.1 Quantitative Analysis

In this section, we use the aforementioned NLL metric to quantitatively evaluate combinations' performance on

Figure 6: MAML-GAN results on Omniglot characters. Upper half contains generated images of MAML-trained GAN adapted on unseen characters for 10 steps; lower half contains generated images of randomly initialized GAN sufficiently trained on unseen characters. Columns 1 and 5 are real images.

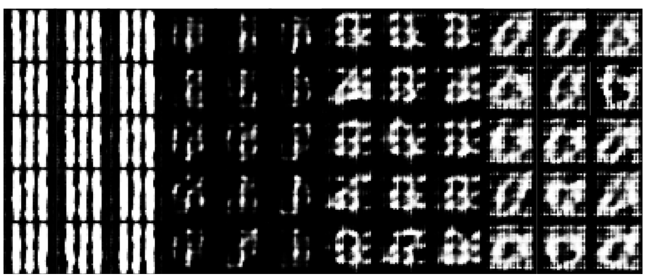


Figure 7: FOMAML-GAN prior generations on MNIST. From left to right are on 5k, 10k, 15k, and 20k meta-updates respectively.

MNIST and Omniglot. For all experiments, we randomly sample 256 times (with replacement) from each meta-task's test tasks. For each test task, the model adapts 10 steps, and NLL is averaged over 4096 generated instances.

Table 1 shows that GAN trained from scratch generates unconvincing results, and the critic makes predictions based on generated images that are no better than random guessing. Interestingly, MetaSGD-GAN has significantly higher NLL than the randomly initialized baseline. This is because meta-SGD generates convincing digits, but generally not of the one of the adaptation task; this behavior can be seen in figure 4. Convincing images causes NLL to be very low for a digit, and in turn causes NLL for the conditioned label to be high. Compared to the baselines, FOMAML-GAN and MAML-GAN offers improvement consistently across NLL and accuracy. Visual results corroborate that they adapt quickly to generate more specific and realistic samples ??.

The quantitative analysis corroborates visual results.

Table 1: Quantitative Results

Algorithm	Task	NLL	Accuracy
Real	MNIST	0.047	0.997
Not pre-trained		15.296	0.099
FIGR		3.195	0.791
MetaSGD-GAN		45.474	0.102
MAML-GAN		2.730	0.823
FOMAML-GAN		**1.449**	**0.881**
Real	Omniglot	0.559	0.949
Not pre-trained		18.894	0.002
FIGR		4.383	0.378
MAML-GAN		**3.919**	**0.392**
FOMAML-GAN		5.286	0.328

(FO)MAML-GAN outperforms both randomly-initialized and reptile baselines in terms of NLL and accuracy, demonstrating our proposed approach's multi-task data generation ability.

6 Conclusion and Future Work

Our work formulates combination of MAML and GAN for multi-task data generation, and investigates the performance of MAML-GAN, FOMAML-GAN, and metaSGD-GAN on MNIST and Omniglot. We used NLL given by pretrained classifier as a quantitative metric to evaluate few-shot generation performance, and report state-of-the-art generalization performance based on this metric.

We took note that revising MAML to optimize multi-step objectives is necessary for training stability and superior performance. While MAML offers stable training and good generation quality, FOMAML offers competitive performance given its losses do not fluctuate greatly; we also offer explanations for the unsatisfactory performance of metaSGD-GAN.

There are several promising directions for future work.

- Apply MAML-GAN, given its competitive performance on MNIST and Omniglot, on larger, more complicated datasets e.g. CIFAR100.
- Use MAML-GAN to enable fast-adaptation in other GAN application scenarios e.g. few-shot style-transfer and few-shot image translation.
- Explore the possibility and results of incorporating batch normalization [15] into MAML-GAN; under small batch sizes, effective batch normalization requires running statistics.
- Conducting few-shot generation experiments on discrete instead of continuous data e.g. multi-style text generation.

References

[1] N. Acar. Is pharmacotherapy effective for the treatment of proliferative vitreoretinopathy? *Retina-Vitreus*, 21(3):166–177, 2016.

[2] N. Acar. Is pharmacotherapy effective for the treatment of proliferative vitreoretinopathy? *Retina-Vitreus*, 21(3):166–177, 2016.

[3] 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 *Advances in Neural Information Processing Systems*, pages 3981–3989, 2016.

[4] Antreas Antoniou, Harrison Edwards, and Amos Storkey. How to train your maml. *arXiv preprint arXiv:1810.09502*, 2018.

[5] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[6] Sergey Bartunov and Dmitry Vetrov. Few-shot generative modelling with generative matching networks. In *International Conference on Artificial Intelligence and Statistics*, pages 670–678, 2018.

[7] Samy Bengio, Yoshua Bengio, Jocelyn Cloutier, and Jan Gecsei. On the optimization of a synaptic learning rule. In *Preprints Conf. Optimality in Artificial and Biological Neural Networks*, pages 6–8. Univ. of Texas, 1992.

[8] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[9] Louis Clouâtre and Marc Demers. Figr: Few-shot image generation with reptile. 2019.

[10] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org, 2017.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[12] Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. Draw: a recurrent neural network for image generation. *Computer Science*, pages 1462–1471, 2015.

[13] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. 2017.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. 2015.

[15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[16] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[17] Brenden Lake, Ruslan Salakhutdinov, Jason Gross, and Joshua Tenenbaum. One shot learning of simple visual concepts. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 33, 2011.

[18] Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-sgd: Learning to learn quickly for few-shot learning. *arXiv preprint arXiv:1707.09835*, 2017.

[19] Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. Few-shot unsupervised image-to-image translation. *arXiv preprint arXiv:1905.01723*, 2019.

[20] Alex Nichol and John Schulman. Reptile: a scalable metalearning algorithm. *arXiv preprint arXiv:1803.02999*, 2, 2018.

[21] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

[22] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *Computer Science*, 2015.

[23] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. 2016.

[24] Scott Reed, Yutian Chen, Thomas Paine, Aäron van den Oord, SM Eslami, Danilo Rezende, Oriol Vinyals, and Nando de Freitas. Few-shot autoregressive density estimation: Towards learning to learn distributions. *arXiv preprint arXiv:1710.10304*, 2017.

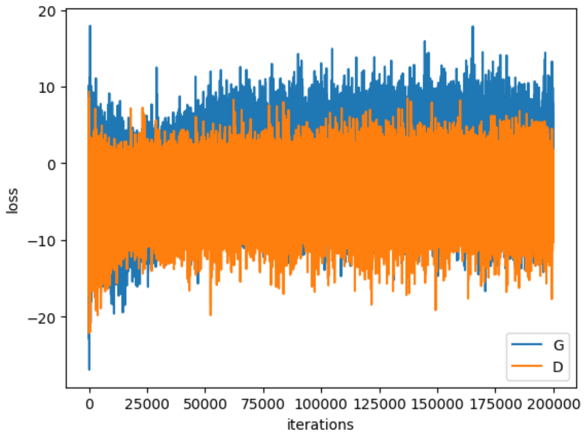
[25] Danilo J Rezende, Shakir Mohamed, Ivo Danihelka, Karol Gregor, and Daan Wierstra. One-shot generalization in deep generative models. 2016.

[26] Jurgen Schmidhuber. Evolutionary principles in self-referential learning. *On learning how to learn: The meta-meta-... hook.) Diploma thesis, Institut f. Informatik, Tech. Univ. Munich*, 1987.

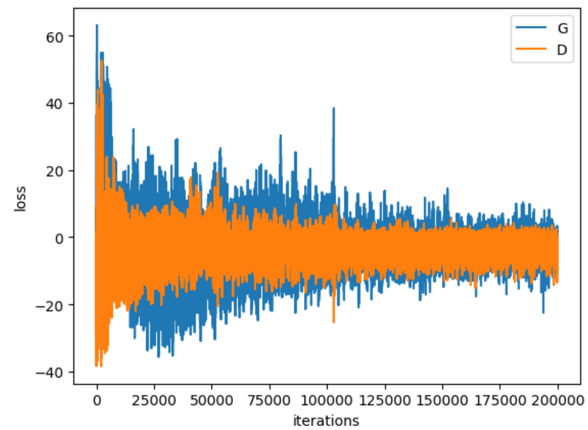
[27] Sebastian Thrun and Lorien Pratt. *Learning to learn*. Springer Science & Business Media, 2012.

[28] Ricardo Vilalta and Youssef Drissi. A perspective view and survey of meta-learning. *Artificial intelligence review*, 18(2):77–95, 2002.

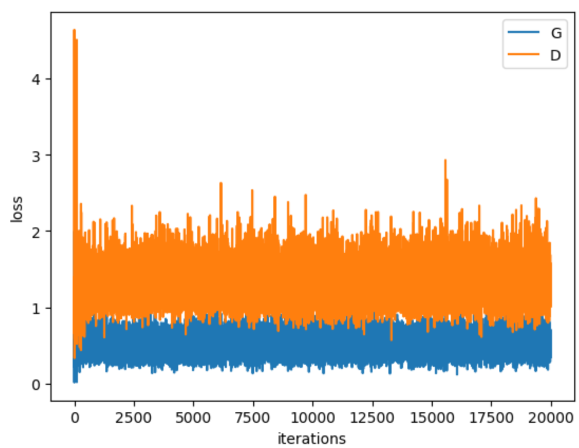
A Training losses



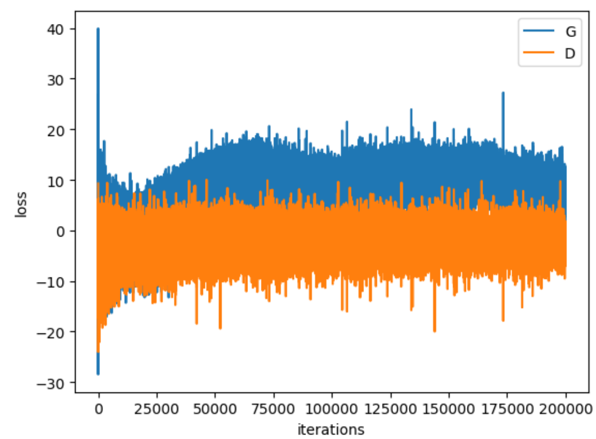
(a) MAML-GAN losses on MNIST



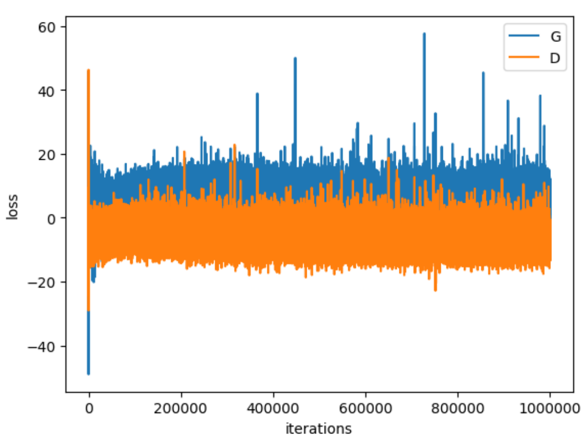
(b) FOMAML-GAN losses on MNIST.



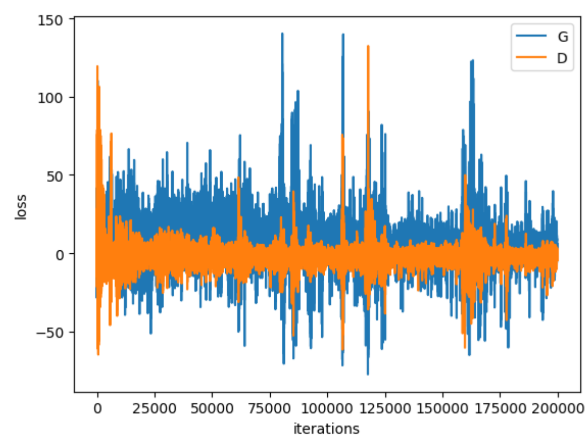
(c) metaSGD-GAN losses on MNIST.



(d) FIGR losses on MNIST.



(e) MAML-GAN losses on Omniglot.



(f) FOMAML-GAN losses on Omniglot.

Figure 8: Learning Curve

致谢：

感谢导师张伟楠教授在论文选题及论文书写规范给予的点拨和帮助与研究过程中给予的鼓励。

选题背景：

2014 年生成对抗网络（GAN）的提出和 2017 年无模型先验元学习（MAML）的提出分别对深度学习界的研究重点产生了很大的影响，MAML 论文也在 2018 年获得 ACM 的最佳论文奖；GAN 展示了深度学习方法在数据生成任务上的能力，而元学习方法大大增加了监督学习任务上的数据和训练效率。在这样的背景下，我们的工作希望尝试通过 MAML 等基于初始化的元学习方法来强化 GAN 的数据和训练效率。

师生关系：

报告作者吕行健在参加 2018 年字节跳动公司（抖音母公司）组织的 AI 夏令营嘉宾授课时认识上海交通大学的张伟楠教授。在夏令营结束后，向张教授请教交流学术问题，从而开始了这个课题。作者每隔两周趁张教授有空的时候会在周二晚上与张教授交流过去两周之内的实验进展并商讨下两周的研究方向。论文 LaTeX 源码全部由作者吕行健完成，张教授在科研及论文撰写过程中无偿进行论文格式及措辞规范的指导。

学术诚信声明：

报告作者吕行健保证，学术报告“Meta-Learning Algorithms for Multi-task Data Generation”由作者在导师张伟楠教授的指导下完成。报告中他人的研究成果尽数在参考文献中注明。

吕行健