Winning Lottery Tickets in Deep Generative Models

Neha Mukund Kalibhat, Yogesh Balaji, Soheil Feizi
Department of Computer Science, University of Maryland - College Park
{nehamk,yogesh,sfeizi}@cs.umd.edu

Abstract

The lottery ticket hypothesis suggests that sparse, sub-networks of a given neural network, if initialized properly, can be trained to reach comparable or even better performance to that of the original network. Prior works in lottery tickets have primarily focused on the supervised learning setup, with several papers proposing effective ways of finding *winning tickets* in classification problems. In this paper, we confirm the existence of winning tickets in deep generative models such as GANs and VAEs. We show that the popular iterative magnitude pruning approach (with late resetting) can be used with generative losses to find the winning tickets. This approach effectively yields tickets with sparsity up to 99% for AutoEncoders, 93% for VAEs and 89% for GANs on CIFAR and Celeb-A datasets. We also demonstrate the transferability of winning tickets across different generative models (GANs and VAEs) sharing the same architecture, suggesting that winning tickets have inductive biases that could help train a wide range of deep generative models. Furthermore, we show the practical benefits of lottery tickets in generative models by detecting tickets at very early stages in training called *early-bird tickets*. Through early-bird tickets, we can achieve up to 88% reduction in floating-point operations (FLOPs) and 54% reduction in training time, making it possible to train large-scale generative models over tight resource constraints. These results out-perform existing early pruning methods like SNIP (Lee, Ajanthan, and Torr 2019) and GraSP (Wang, Zhang, and Grosse 2020). Our findings shed light towards existence of proper network initializations that could improve convergence and stability of generative models.

Introduction

The lottery ticket hypothesis (Frankle and Carbin 2018), suggests that there exist sparse sub-networks in over-parameterized neural networks that can be trained to achieve similar or better accuracy than the original network, under the same parameter initialization. These sub-networks and their associated initializations form *winning tickets*. In addition to saving memory, winning tickets have also been shown to achieve improved performance (Frankle and Carbin 2018).

Evidence of the existence of winning tickets has been shown successfully on Visual Recognition tasks (on various CNN-based architectures such as VGG and ResNet) (Morcos et al. 2019), Reinforcement Learning and Natural Language Processing tasks (Yu et al. 2020). While research in finding lottery tickets has primarily focused on the classification problem, to the best of our knowledge, no prior work exists on understanding the lottery ticket hypothesis in deep generative models. This will be the focus of our work.

In particular, we investigate if winning tickets exist in two popular families of deep generative models — Variational AutoEncoders (VAEs) (Kingma and Welling 2014) and Generative Adversarial Networks (GAN) (Goodfellow et al. 2014). VAEs are relatively easier to train compared to GANs since their optimization involves a single minimization objective (i.e. the evidence lower bound). Training GANs, on the other hand, is notoriously difficult as it involves a *min-max* game between the generator and the critic networks, causing instability in training (Goodfellow et al. 2014; Arjovsky, Chintala, and Bottou 2017). In both GANs and VAEs, models are in the over-parameterized regime (Brock, Donahue, and Simonyan 2018; Razavi, van den Oord, and Vinyals 2019) to improve the quality of reconstructions and sample generations, especially in large-scale datasets such as Celeb-A (Liu et al. 2015) and ImageNet (Deng et al. 2009). This results in significant overhead in training time, compute and memory requirements. Thus, it will be useful, in terms of model storage, training time and training stability, to address the fundamental issue of over-parameterization in deep generative models.

Another motivation is to see if winning tickets can be *transferred* across different generative models. In particular, the generator network in GANs and the decoder network in VAEs fundamentally perform the same task, i.e. transforming the input vectors in the latent space to realistic images. This hints that winning tickets (sub-networks and their initializations) on one generative model (e.g. VAE) might be also a successful ticket on another generative model (e.g. GAN) of the same architecture, although they are based on completely different loss functions. If this is indeed the case, this would imply that a wide range of generative models with different loss functions can share similar network structures and initializations, showing generalizability and scalability of winning tickets. Therefore, to verify transferability of winning tickets, we train the generator of a GAN using the winning ticket obtained from the decoder of VAE, and vice

Copyright © 2021, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

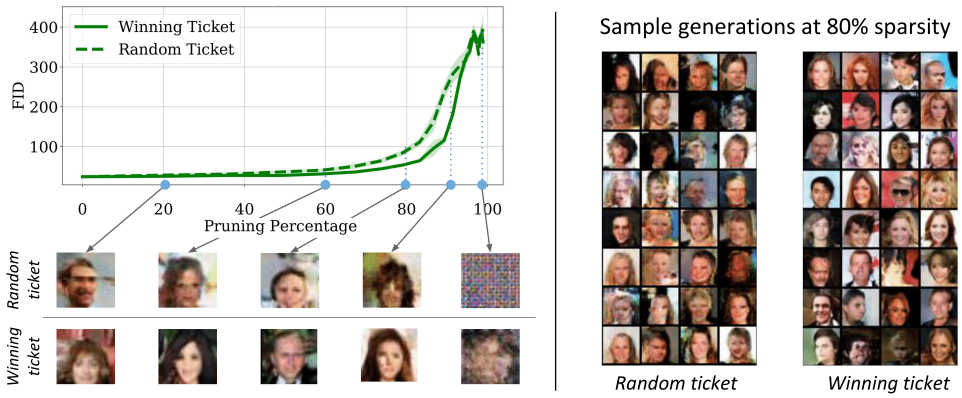

Figure 1: Lottery Ticket Hypothesis in Generative Models. The panel on the left shows FID scores of winning tickets and random tickets on a DCGAN model trained on Celeb-A. Winning tickets clearly outperform random tickets at higher pruning regimes. The improved performance of winning tickets is also evident from qualitative results where we find winning tickets generate better quality samples at all sparsity levels. The panel on the right shows sample generations at 80% sparsity.

versa, while keeping the other components of both models unpruned.

The process of finding winning tickets uses a technique called *Iterative Magnitude Pruning*, which involves alternating between network pruning and network re-training steps, while gradually pruning the model. At each iteration of this process, we obtain a sparse sub-network along with its parameter initializations, both of which constitute a *ticket*. We also observe that *late rewinding* (Frankle et al. 2019) is a favorable approach in generative models. Therefore, for each model, we compare the performance of (1) winning ticket, and (2) randomly-initialized ticket. If the randomly initialized sub-network performs significantly worse, a *winning ticket* is obtained. AutoEncoders and VAEs consist of two components: an encoder and a decoder network, while GANs have a generator and a discriminator. In each case, we perform iterative pruning experiments on either jointly pruning both networks of the model or by pruning each network separately while leaving the other unpruned.

In contrast to classification tasks where there is a well-defined evaluation metric (i.e. classification accuracy) for assessing the model performance, we do not have such a metric in deep generative modeling. On image-based datasets, Fréchet Inception Distance (FID) (Heusel et al. 2017) is one popular metric used in evaluating deep generative models. Figure 1 shows winning and random tickets evaluated on FID and also illustrates the differences in generated images. We have also used metrics such as the reconstruction loss (in AutoEncoders), the discriminator loss (in GANs) and the downstream classification accuracy (in AutoEncoders and VAEs).

The existence of winning tickets indicates that generative models can be trained under limited memory and resource constraints. However, finding these tickets requires multiple rounds of training, and each training cycle can last for weeks for large models such as BigGAN((Brock, Donahue, and Simonyan 2018)). Moreover, the sparse sub-networks found using iterative magnitude pruning have individual parameter-level sparsity (as opposed to channel-level sparsity). Hence, minimizing compute is not possible without using specialized hardware (Cerebras 2019; NVIDIA 2020). Instead, we are interested in finding better pruning strategies that provides compute gains on existing hardware. To this end, we investigate the effectiveness of *early-bird tickets*, which are channel-pruned sub-networks found early in the training (You et al. 2020) in the context of generative models. We also compare the performance of early-bird tickets with other pruning strategies like SNIP (Lee, Ajanthan, and Torr 2019) and GraSP (Wang, Zhang, and Grosse 2020) that prune the network at initialization.

We conduct experiments on several generative models including linear AutoEncoder, convolutional AutoEncoder, VAE, β-VAE (Higgins et al. 2017), ResNet-VAE (Kingma et al. 2016), Deep-Convolutional GAN (DCGAN) (Radford, Metz, and Chintala 2015), Spectral Normalization GAN (SNGAN) (Miyato et al. 2018), Wasserstein GAN (WGAN) (Arjovsky, Chintala, and Bottou 2017) and ResNet-GAN (He et al. 2016) on MNIST (LeCun, Cortes, and Burges 2010), CIFAR-10 (Krizhevsky 2009) and Celeb-A (Liu et al. 2015) datasets. Table 1 summarizes all our experiments and the winning ticket sparsity achieved for each model. We make the following observations:

- **AutoEncoders:** We find winning tickets of sparsity 89% on MNIST, 96% on CIFAR-10 and 99% on Celeb-A.
- **VAEs:** We find winning tickets of sparsity 79% on VAE, 87% on β-VAE and 93% on ResNet-VAE on both datasets.
- **GANs:** We find winning tickets of sparsity 83% on DCGAN, 89% on SNGAN and 79% on ResNet-DCGAN on both datasets. In WGAN we find winning tickets at 73.7% (CIFAR-10) and 59% (Celeb-A).
- **Single Component Pruning:** In AutoEncoders and GANs, comparable sparsities of both components is essential for the best model performance. In VAEs, on the other hand, encoder-only pruning preserves performance, implying that VAEs can be trained with very sparse en-

	Network	Number of Parameters	Datasets	Winning Ticket Sparsity	Evaluation Metric
AE	Linear AutoEncoder	200K	MNIST	89.2%	Reconstruction Loss, Test Accuracy
	Conv. AutoEncoder	3M	CIFAR-10	95.6%	
			Celeb-A	98.5%	
VAE	VAE (Kingma and Welling 2014)	5.6M	CIFAR-10, Celeb-A	79%	FID, Test Accuracy
	β-VAE (Higgins et al. 2017)	5.6M		86.5%	
	ResNet-VAE (Kingma et al. 2016)	2.8M		93.1%	
GAN	DCGAN (Radford, Metz, and Chintala 2015)	6.5M	CIFAR-10, Celeb-A	83.2%	FID, Discriminator Loss
	SNGAN (Miyato et al. 2018)	6.7M		89.2%	
	ResNet-DCGAN	2.2M		79%	
	WGAN (Arjovsky, Chintala, and Bottou 2017)	6.5M	CIFAR-10	73.7%	
			Celeb-A	59%	

Table 1: Summary of lottery ticket experiments conducted on various generative models

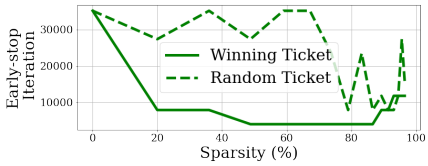

Figure 2: The plot shows the early-stopping iteration of lottery tickets. Winning lottery tickets show significantly faster convergence than random tickets. Experiments are performed on 5 random runs, and the error bars represent $+/-$ standard deviation across runs.

coders.

- **Late Rewinding:** For very deep generative models, we observe that rewinding weights to an early iteration instead of initialization, is favorable for better stability and performance of winning tickets.
- **Stability:** Winning tickets demonstrate higher stability across multiple runs compared to random tickets.
- **Convergence:** Winning tickets converge significantly faster than the unpruned network and random tickets. This is demonstrated in Figure 2.
- **Transferability:** Winning tickets transferred from VAEs to GANs perform on par to winning tickets solely found on GANs and vice versa.
- **Early-Bird Tickets:** Early-Bird tickets reduce the training time by 54% and FLOPs by 88%. They outperform other early pruning strategies like SNIP (Lee, Ajanthan, and Torr 2019) and GraSP (Wang, Zhang, and Grosse 2020) in terms of FID, FLOPs and training time.

These results shed some light on existence of proper network initializations and architectures in deep generative models that could improve their computational and statistical properties such as their convergence, stability and storage. In particular, the transferability of winning tickets across seemingly different generative models such as GANs and VAEs suggest that there exists unifying network architectures and initialization strategies across these models.

Related Work

Generative Models. Two prominent and popular deep generative models are VAEs and GANs. VAEs train a generative model by maximizing an Evidence Lower Bound (ELBO) with an encoder-decoder structure, in which the encoder maps the data samples to a latent space, while the decoder reconstructs the latent representations back to the input space (Kingma and Welling 2014). β-VAE (Higgins et al. 2017) proposes a modification to the VAE objective, where an adjustable hyper-parameter β balances the reconstruction accuracy and latent representation constraints, leading to improved qualitative performance. On the other hand, GANs (Goodfellow et al. 2014) train a generative model by transforming input samples with known tractable distribution (such as Gaussian) to an unknown data distribution (e.g. images). This transformation function is learnt using an adversarial game between generator and discriminator networks. This *min-max* game results in difficulties in optimization, especially in deep networks. To improve the stability, several techniques such as spectral normalization (Miyato et al. 2018), Wasserstein losses (Arjovsky, Chintala, and Bottou 2017), gradient penalties (Gulrajani et al. 2017), self-attention models (Zhang et al. 2018), etc. have been proposed.

Generative Model Pruning. Network pruning and model compression are important topics in machine learning especially in supervised learning setups (Cun, Denker, and Solla 1990; Hassibi et al. 1993; Han et al. 2016; Li et al. 2016). However, these problems have been relatively less explored in deep generative models. The magnitude pruning approach used in this paper (Han et al. 2015), zeroes out weights with small magnitudes, followed by re-training. Other pruning approaches include pruning groups of weights together (i.e. structured sparsity learning) (Wen et al. 2016) , pruning filters and their connecting edges (Li et al. 2016) and enforcing channel-level sparsity (Liu et al. 2017). SNIP (Lee, Ajanthan, and Torr 2019) and (Wang, Zhang, and Grosse 2020) are recent approaches that propose pruning at initialization. Knowledge distillation-based approaches (Hinton, Vinyals, and Dean 2015), in which smaller networks are trained using the distillation loss from a larger teacher network have also been successfully used in compressing GANs (Aguinaldo et al. 2019; Koratana et al. 2018).

Lottery Ticket Hypothesis. The lottery ticket hypothesis, (Frankle and Carbin 2018) proposes Iterative Magnitude Pruning (IMP) to find tickets. In deeper networks, it

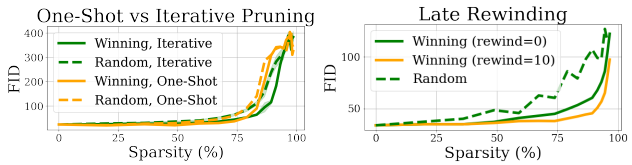

Figure 3: The first plot shows the comparison of one-shot and iterative pruning on DCGAN trained on CIFAR-10. The second plot shows FID of tickets rewound to initialization and to iteration 10. Iterative pruning and late rewinding shows better tickets at high sparsities. Experiments are performed on 5 random runs, and the error bars represent $+/-$ standard deviation across runs.

has been shown that IMP with late-rewinding (Frankle et al. 2019; Morcos et al. 2019) is more beneficial than rewinding to initialization. While many early pruning strategies (Lee, Ajanthan, and Torr 2019; Wang, Zhang, and Grosse 2020) have been proposed, it has been shown that these methods often fall-short compared to IMP (Frankle et al. 2020). It is also shown (Zhou et al. 2019) that signs of the weights are more important than their magnitudes and as long as they remain the same, the sparse model can still train more successfully than with random sign assignment initializations. However, a somehow contradicting observation is made (Frankle, Schwab, and Morcos 2020), where authors show that deep networks are not robust to random weights while maintaining signs. It has also been shown that winning tickets reflect inductive biases and do not over-fit to particular domains (Desai, Zhan, and Aly 2019). Although the initial work on the hypothesis has focused on supervised image classification tasks, successful results on other tasks have been observed too. Transfer learning tasks (Mehta 2019), transformer models in Natural Language Processing and Reinforcement Learning (Yu et al. 2020) have successfully uncovered winning tickets. The lottery ticket hypothesis however, has been challenged in (Liu et al. 2018) which argues that randomly initialized tickets can match performance to winning tickets if trained with an optimal learning rate and for long enough.

Methods

Pruning Approach

Winning Lottery Tickets One-shot pruning and Iterative Magnitude Pruning are standard approaches to find winning lottery tickets. In one-shot pruning, the lower $p\%$ weights of the trained network with initialization θ_0 (i.e. parameters with the smallest magnitudes) are pruned and the remaining weights are re-initialized to θ_0 and re-trained. In practice however, pruning a large fraction of weights through one-shot pruning might null the weights that are actually important to the model leading to a significant drop in the performance (Morcos et al. 2019). In Figure 3, we confirm that this phenomenon applies to generative models as well.

We instead employ *Iterative Magnitude Pruning* (IMP). Here, we take a trained model initialized with θ_0 and we choose a small pruning percentage p. At the first pruning cycle, we one-shot prune $p\%$ of network, generating a mask m_1 and re-train the network using (θ_0, m_1). In the next pruning cycle, $p\%$ of the remaining weights from the previous cycle are one-shot pruned (generating m_2) and re-trained with (θ_0, m_2). We repeat this process of pruning and re-training with masks for n rounds. In this paper, we use a global pruning scheme across all experiments; i.e. weights of all the layers of the network are pooled together and pruned. In all our experiments in this paper, $p = 20\%$ and $n = 20$, i.e. we run 20 rounds of iterative magnitude pruning where we prune 20% of the network at each iteration. Since p is not too large, it properly scans pruning fractions between 20% to 98.84%.

It has been shown that rewinding the network to the weights at training iteration i, θ_i (where $i \ll N$, N being the total number of training iterations) is better than rewinding to θ_0 (Frankle et al. 2019) as deep neural networks become more stable to noise after a few iterations of training. We confirm this behavior for generative models in Figure 3. Therefore, unless specified otherwise, we apply late-rewinding to all our experiments.

Early-bird Tickets Winning tickets found using IMP is an expensive process involving multiple training cycles for finding the optimal ticket. This is impractical in real-world applications, especially for generative models. In the recent work of (You et al. 2020), it has been shown that lottery tickets emerge at very early stages of training, also termed as *early-bird tickets* (EB-tickets). EB-tickets reduce the overhead of iterative pruning by finding the ticket early, without having to train models to convergence. Once identified, training can be continued on just the EB-tickets towards convergence.

The process of finding EB-tickets involves using channel pruning, in which a fraction of batch-normalization channels of the network are pruned (You et al. 2020). While magnitude pruning removes individual parameters from a neural net, channel pruning effectively removes the entire channel, since setting the affine batch normalization parameters has the same effect of removing the corresponding channel it connects to. This pruning technique results in a shallower network and yields savings in memory and compute without any additional hardware requirements. At every training iteration i, we perform channel pruning to get a mask m_i. We then compute the *mask distance*, defined as the Hamming distance between m_i and m_{i-1} (look-back can be over multiple iterations). When the mask distance is less than an upper bound δ, an EB-ticket is found. We then compress the network channels and continue training the EB-ticket. In our experiments, we look-back 5 iterations and fix δ as 0.1. These hyper-parameters generally help us find stable EB-tickets very early in training at epoch 4 to 6. We also perform mixed-precision training on EB-tickets, where the floating-point precision of parameters and their gradients are reduced from 32-bit to 16-bit or 8-bit depending on their sizes.

Evaluation

Winning Lottery Tickets Tickets discovered in the iterative pruning process need to be tested to verify if the cause of their good performance is their initialization. For such an

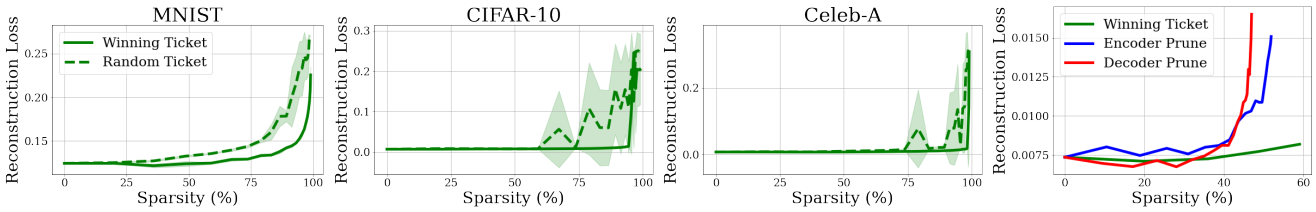

Figure 4: We plot reconstruction losses of tickets at different levels of sparsity on MNIST, CIFAR-10 and Celeb-A. In all datasets, performance of winning tickets is consistently better than random tickets. Each experiment is performed on 5 random runs, and the error bars represent $+/-$ standard deviation across runs.

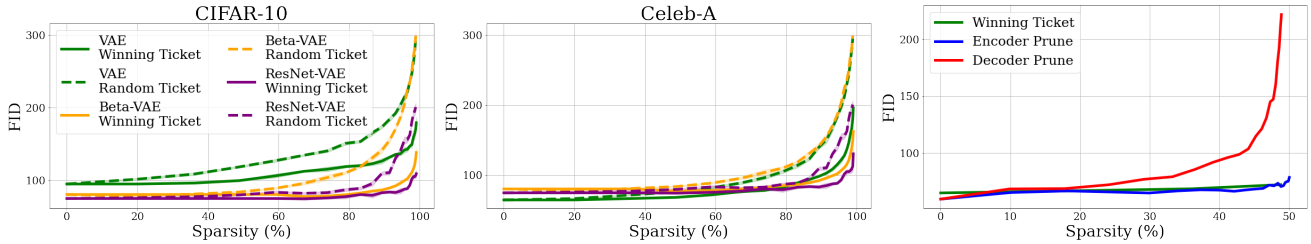

Figure 5: We plot the FID scores of tickets on CIFAR-10 and Celeb-A on three models: VAE, β-VAE, and ResNet-VAE. Winning tickets outperform random tickets on all models. Experiments are performed on 5 random runs, and the error bars represent $+/-$ standard deviation across runs.

evaluation, we measure the performance of winning tickets when they are randomly initialized called, *random tickets*.

For generative models, we use the following metrics: (1) For AutoEncoders, we use the reconstruction loss after the model has converged. We also measure downstream classification accuracy, in which we pass the reconstructed images to a ResNet-18 model trained on CIFAR-10 and Celeb-A, and calculate the test accuracy on the reconstructed samples. (2) In VAEs, we use FID to assess the quality of generated samples. The FID score (Heusel et al. 2017) calculates the Fréchet Inception distance between the feature distributions (as given by a pre-trained Inception network) of real and generated samples. These feature distributions are approximated using a Gaussian distribution. Then, for two distributions p_r, p_g with feature means and co-variances (μ_r, Σ_r) and (μ_g, Σ_g) we calculate

$$FID(p_r, p_g) = \|\mu_r - \mu_g\|^2 + Tr(\Sigma_r + \Sigma_g) - 2(\Sigma_r\Sigma_g)^{\frac{1}{2}}$$

A small change in FID may show little or no change in the quality of the generated samples. However, a large change in FID (e.g. a change $\gg 10$) translates to a perceptible difference in the quality of generated samples. We also use the downstream classification test accuracy as an additional evaluation metric since FID scores in VAEs are often high due to the blurriness of generated images. (3) In GANs, we evaluate winning tickets using FID and the discriminator loss. In addition to these quantitative metrics, we also evaluate images generated by the winning tickets qualitatively. Winning tickets should preserve and generate higher quality images than random tickets. Table 1 summarizes the evaluation metrics used for each generative model.

Early-bird Tickets We evaluate early-bird tickets in generative models, similar to winning lottery tickets using FID as discussed in the previous section. In addition to FID, we measure the resource usage both in terms of: (1) Floating-point operations (FLOPs) and (2) Training Time. FLOPs are measured by cumulatively adding the floating-point operations of forward and backward propagations of convolutional, linear and batch-normalization layers over the entire training cycle, including pruning operations, giving us an exact measure of the total processor usage. In addition to FLOPs, we also measure the total training time in seconds.

Results

Winning Tickets in AutoEncoders

In this section, we discuss the the winning tickets in AutoEncoders by evaluating the reconstruction loss with random tickets. In Figure 4, we observe that the winning tickets of a Linear AutoEncoder on MNIST preserve the reconstruction loss up to a sparsity of 89%. The same tickets, when randomly initialized, perform visibly worse. In Convolutional AutoEncoders, we achieve winning ticket sparsity up to 96% in CIFAR-10 and 99% in Celeb-A. Note that a sparsity of 99% reduces the number of weights from 3 million to around only 30K.

It is also interesting to see how the winning ticket training curves are more well-behaved across runs compared to those of randomly-initialized tickets especially at higher pruning percentages. This indicates that good initializations do help stabilize the training process. Finally, the images reconstructed by the winning tickets maintain high quality, thus validating the presence of lottery tickets in AutoEncoders

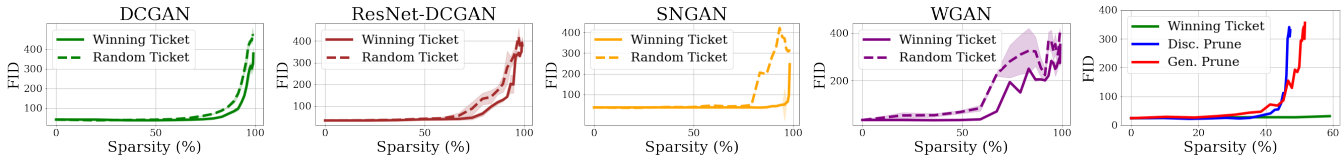

Figure 6: We plot the FID scores of tickets of 4 GAN models trained on CIFAR-10 dataset: DCGAN, ResNet-GAN, SNGAN and WGAN. Winning tickets consistently outperform random tickets on all models. Experiments are performed on 5 random runs, and the error bars represent $+/-$ standard deviation across runs.

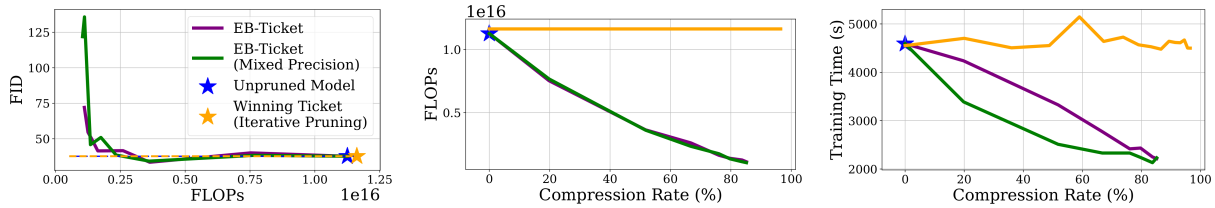

Figure 7: In the first plot, we compare the FID of tickets against FLOPs (floating-point operations). In the second plot, we see the change in FLOPs at different compression rates. EB-tickets show a significant reduction in FLOPs while maintaining performance. In the last plot, we see the change in training time of tickets at different compression rates. Mixed precision EB-tickets train the fastest. Experiments are performed on 5 random runs, and the error bars represent $+/-$ standard deviation across runs.

(See Appendix for more details).

In Figure 4, we also demonstrate how differently the AutoEncoder behaves when we prune both components (Winning Ticket, green curve) and when a single component is pruned (blue, red curves). We observe that winning tickets with both components pruned, outperform single-component-pruned tickets. This implies that comparable network parameters in the encoder and decoder of AutoEncoders are essential for best results.

Winning Tickets in Variational AutoEncoders

The presence of winning tickets in AutoEncoders provides a strong indication that they could exist in VAEs as well. In VAEs, we observe that a sparsity of 79% preserves FID in both CIFAR-10 and Celeb-A as shown in Figure 5. We also show that winning tickets are not restricted to the loss function or the architecture of the VAE. Winning tickets are visible in β-VAEs at 87% sparsity and on ResNet-VAE at 93% sparsity on both CIFAR-10 and Celeb-A. Winning tickets are also visible when evaluated on the classification accuracy and generated images (See Appendix for more details). It is evident that the ResNet-VAE shows winning tickets of highest quality in terms of FID, accuracy and generated images., effectively reducing the model size from 2.8 million to around only 196K.

Figure 5 also shows us the behavior of the VAE when single components are pruned. Note that we can only achieve around 50% network sparsity when a single component of the network is pruned, since the other component accounts for the other 50%. We observe that pruning the encoder shows negligible change in FID and is almost aligned with the winning ticket performance even up to 50% network

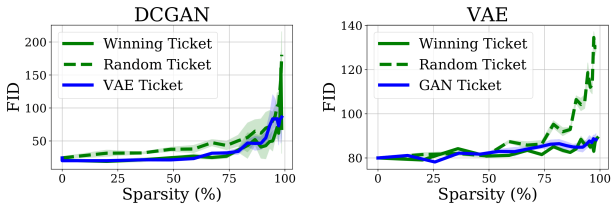

Figure 8: The plot on the top compares DCGAN winning tickets against the VAE winning ticket (transfer ticket) trained on DCGAN. The performance of the transfer ticket is comparable to that of GAN's own winning ticket. The plot on the bottom shows the other direction where VAEs are trained with tickets obtained from the GAN. Performance of transfer ticket is on-par with VAE's own ticket. Experiments are performed on 5 random runs, and the error bars represent $+/-$ standard deviation across runs.

sparsity. Decoder pruning performs significantly worse than encoder pruning and the winning ticket. Our observation indicates that VAE can be trained with extremely sparse encoders, without affecting its performance.

Winning Tickets in Generative Adversarial Networks

VAEs and AutoEncoders are both unsupervised models and their optimization objectives, based on minimizing the reconstruction and ELBO, is a minimization problem resembling that of the supervised prediction models. On the other hand, GANs are formulated as a min-max optimization

problem which fundamentally differs from single minimization problems. In this section, we desire to see whether or not winning tickets exist in such generative models that are optimized game-theoretically.

In Figure 6, we see that tickets are seen at 83% sparsity in DCGAN, 89% in SNGAN and 79% in ResNet-DCGAN. WGAN shows winning tickets of sparsity 73.7% on CIFAR-10 and 59% on Celeb-A (See Appendix for more details). The best performing winning tickets bring the size of the network from 6.7 million parameters to nearly 737K. We also see winning tickets when we evaluate the discriminator loss and the quality of generated images (See Appendix for more details). These results confirm the lottery ticket hypothesis in GANs under different loss functions, architectures and evaluation metrics.

We also show in Figure 6 that, similar to AutoEncoders, winning tickets when both components are pruned (green curve) outperform tickets with single component pruning (blue, red curves). With this observation, we conclude that the generator and discriminator in GANs should be of comparable sizes for GANs to perform well, even under very sparse regimes.

Transferability of Winning Tickets from VAEs to GANs

In this section, we show evidence of transferability of winning tickets across generative models. We know that the VAE's decoder network and GAN's generator network share the same architecture and task. Therefore, we transfer VAE's decoder winning tickets and train them under a DCGAN generator setup while keeping the discriminator unpruned. The first plot in Figure 8 compares three cases: (1) The green and (2) dashed green curves represent winning and random tickets found by pruning the DCGAN generator. (2) The blue curve shows the behavior of winning tickets when transferred from the VAE's decoder to the DCGAN's generator. We see that the VAE-initialized winning tickets match the performance of the tickets found using GANs, and preserve performance up to 80%. We observe similar results when we transfer winning tickets from GAN's generator to the VAE decoder where the transferred tickets behave comparably to the VAE winning tickets. This shows that a single initialization succeeds in training winning tickets in both networks. Hence, this confirms our hypothesis that winning tickets can be transferred across different deep generative models and provides evidence for a universal weight initialization that could work well across a range of generative models.

Early-Bird Tickets

In this section, we observe the behavior of early-bird tickets and compare it with winning lottery tickets found using IMP. In Figure 7, (1) the purple line represents EB-tickets at different network compression rates ranging from 20% to 95%, (2) the green line represents the same EB-tickets under mixed-precision training, (3) the yellow star/line represents winning lottery tickets found in the previous sections and (4) the blue star represents the unpruned network.

Pruning Technique	FID	Number of weights	FLOPs	Training Time (seconds)
Unpruned Network	37.71	5651584	1.12e+16	4590.5
Iterative Magnitude Pruning	37.77	5651584	1.16e+16	4477.7
EB-Ticket	**33.49**	**1148190**	**0.13e+16**	**2417.07**
EB-Ticket (mixed-precision)	34.28	1148190	0.13e+16	2131.09
SNIP (Lee, Ajanthan, and Torr 2019)	56.02	5651584	1.12e+16	4689.5
GraSP (Wang, Zhang, and Grosse 2020)	65.53	5651584	1.12e+16	4603.3

Table 2: Comparing the performance of Early-Bird Tickets in DCGAN to other early pruning techniques

We observe that, in DCGAN, the training FLOPs can be reduced from 11.2 quadrillion (unpruned network) to 1.3 quadrillion (over 88% reduction), with a negligible change in FID. Mixed-precision trained EB-tickets perform on par with full-precision trained EB-tickets in terms of FLOPs. The training time of mixed-precision EB-tickets (53.5% reduction), however, is better than full-precision tickets (47% reduction). This indicates that reduced-precision training is a simple strategy that can reduce memory and computation without compromising on the model performance.

The winning lottery tickets, on the other hand, show an increase in FLOPs compared to the unpruned network due to repetitive masking of the network after every iteration. The FLOPs and training time for winning lottery tickets are also consistently high across all sparsities, while EB-tickets consistently reduce FLOPs and training time as the compression rate increases.

Finally, we compare the performance of EB-tickets to other recent early pruning strategies like SNIP (Lee, Ajanthan, and Torr 2019) and GraSP (Wang, Zhang, and Grosse 2020) in Table 2. SNIP and GraSP are sparse pruning strategies that prune the network at initialization. Therefore, although they prune networks very early in training, they show no reduction in FLOPs, training time and number of weights. More importantly, they are unfavorable for generative models as they do not produce good FIDs. The EB-tickets therefore, outperform SNIP and GraSP in every aspect. These results also align with recent work (Frankle et al. 2020) that shows pruning at initialization is always inadequate.

Conclusion

The key finding in this paper is that the Lottery Ticket hypothesis holds in deep generative models such as VAEs and GANs under different loss functions and architectures. We show that these winning tickets are visible under multiple evaluation metrics. We confirm that winning tickets can be transferred across generative models with different objective functions indicating that a single initialization can successfully train multiple generative models to convergence. Finally, with early-bird tickets we show the most effective and practical approach to train generative models using significantly lesser resources. Thus, large generative models can be optimized using lottery tickets with improved training time, storage and computation resources. Applying our findings on even larger GANs like BigGAN (Brock, Donahue, and Simonyan 2018), is a direction for future research.

Acknowledgements

This project was supported in part by NSF CAREER AWARD 1942230 and a Simons Fellowship on Deep Learning Foundations.

Broader Impact

Deep generative models are used for a variety of tasks such as image generation, image editing, 3D object generation, video prediction and image in-painting. Powerful GANs that perform such tasks on large-scale datasets such as ImageNet (Deng et al. 2009), require TPUs of 128 to 512 cores to generative high quality images. However, in reality, users capture, store and interact with images on mobile phones which have a limited compute power. Several learning tasks as mentioned above, therefore, become too far-fetched to accomplish in real-time on end-user devices. Our work opens up a possibility of training deep generative models without the requirement of powerful GPUs and large amounts of memory, thus potentially opening their applications to a broader community of users. We thus shift the focus to powerful initializations of small networks to achieve improved results. Finally, to the best of our knowledge, this work will not create any negative societal or ethical impacts.

References

Aguinaldo, A.; Chiang, P.-Y.; Gain, A.; Patil, A.; Pearson, K.; and Feizi, S. 2019. Compressing GANs using Knowledge Distillation.

Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein GAN.

Brock, A.; Donahue, J.; and Simonyan, K. 2018. Large Scale GAN Training for High Fidelity Natural Image Synthesis.

Cerebras. 2019. Cerebras wafer scale engine: An introduction, 2019. https://www.cerebras.net/wp-content/uploads/2019/08/Cerebras-Wafer-Scale-Engine-An-Introduction.pdf.

Cun, Y. L.; Denker, J. S.; and Solla, S. A. 1990. Optimal Brain Damage. In *Advances in Neural Information Processing Systems*, 598–605. Morgan Kaufmann.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Desai, S.; Zhan, H.; and Aly, A. 2019. Evaluating Lottery Tickets Under Distributional Shifts. *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)* doi:10.18653/v1/d19-6117. URL http://dx.doi.org/10.18653/v1/d19-6117.

Frankle, J.; and Carbin, M. 2018. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks.

Frankle, J.; Dziugaite, G. K.; Roy, D. M.; and Carbin, M. 2019. Linear Mode Connectivity and the Lottery Ticket Hypothesis.

Frankle, J.; Dziugaite, G. K.; Roy, D. M.; and Carbin, M. 2020. Pruning Neural Networks at Initialization: Why are We Missing the Mark?

Frankle, J.; Schwab, D. J.; and Morcos, A. S. 2020. The Early Phase of Neural Network Training.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A. C.; and Bengio, Y. 2014. Generative Adversarial Networks. *ArXiv* abs/1406.2661.

Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; and Courville, A. 2017. Improved Training of Wasserstein GANs.

Han, S.; Pool, J.; Narang, S.; Mao, H.; Tang, S.; Elsen, E.; Catanzaro, B.; Tran, J.; and Dally, W. 2016. DSD: Regularizing Deep Neural Networks with Dense-Sparse-Dense Training Flow .

Han, S.; Pool, J.; Tran, J.; and Dally, W. J. 2015. Learning both Weights and Connections for Efficient Neural Network. *ArXiv* abs/1506.02626.

Hassibi, B.; Stork, D. G.; Wolff, G.; and Watanabe, T. 1993. Optimal Brain Surgeon: Extensions and Performance Comparisons. In *Proceedings of the 6th International Conference on Neural Information Processing Systems*, NIPS'93, 263–270. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* doi:10.1109/cvpr.2016.90. URL http://dx.doi.org/10.1109/cvpr.2016.90.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium.

Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M. M.; Mohamed, S.; and Lerchner, A. 2017. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In *ICLR*.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the Knowledge in a Neural Network.

Kingma, D. P.; Salimans, T.; Jozefowicz, R.; Chen, X.; Sutskever, I.; and Welling, M. 2016. Improving Variational Inference with Inverse Autoregressive Flow.

Kingma, D. P.; and Welling, M. 2014. Auto-Encoding Variational Bayes. *CoRR* abs/1312.6114.

Koratana, A.; Kang, D.; Bailis, P.; and Zaharia, M. 2018. LIT: Block-wise Intermediate Representation Training for Model Compression.

Krizhevsky, A. 2009. Learning multiple layers of features from tiny images. Technical report.

LeCun, Y.; Cortes, C.; and Burges, C. 2010. MNIST handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist* 2.

Lee, N.; Ajanthan, T.; and Torr, P. 2019. SNIP: SINGLE-SHOT NETWORK PRUNING BASED ON CONNECTION SENSITIVITY. In *International Conference on Learning Representations*. URL https://openreview.net/forum?id=B1VZqjAcYX.

Li, H.; Kadav, A.; Durdanovic, I.; Samet, H.; and Graf, H. P. 2016. Pruning Filters for Efficient ConvNets.

Liu, Z.; Li, J.; Shen, Z.; Huang, G.; Yan, S.; and Zhang, C. 2017. Learning Efficient Convolutional Networks through Network Slimming. *2017 IEEE International Conference on Computer Vision (ICCV)* doi:10.1109/iccv.2017.298. URL http://dx.doi.org/10.1109/ICCV.2017.298.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep Learning Face Attributes in the Wild. In *Proceedings of International Conference on Computer Vision (ICCV)*.

Liu, Z.; Sun, M.; Zhou, T.; Huang, G.; and Darrell, T. 2018. Rethinking the Value of Network Pruning.

Mehta, R. 2019. Sparse Transfer Learning via Winning Lottery Tickets.

Miyato, T.; Kataoka, T.; Koyama, M.; and Yoshida, Y. 2018. Spectral Normalization for Generative Adversarial Networks.

Morcos, A. S.; Yu, H.; Paganini, M.; and Tian, Y. 2019. One ticket to win them all: generalizing lottery ticket initializations across datasets and optimizers. In *NeurIPS*.

NVIDIA. 2020. Nvidia a100 tensor core gpu architecture, 2020. https://www.nvidia.com/content/dam/en-zz/Solutions/Data-Center/nvidia-ampere-architecture-whitepaper.pdf.

Radford, A.; Metz, L.; and Chintala, S. 2015. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks.

Razavi, A.; van den Oord, A.; and Vinyals, O. 2019. Generating Diverse High-Fidelity Images with VQ-VAE-2.

Wang, C.; Zhang, G.; and Grosse, R. 2020. Picking Winning Tickets Before Training by Preserving Gradient Flow. In *International Conference on Learning Representations*. URL https://openreview.net/forum?id=SkgsACVKPH.

Wen, W.; Wu, C.; Wang, Y.; Chen, Y.; and Li, H. 2016. Learning Structured Sparsity in Deep Neural Networks.

You, H.; Li, C.; Xu, P.; Fu, Y.; Wang, Y.; Chen, X.; Baraniuk, R. G.; Wang, Z.; and Lin, Y. 2020. Drawing Early-Bird Tickets: Toward More Efficient Training of Deep Networks. In *International Conference on Learning Representations*. URL https://openreview.net/forum?id=BJxsrgStvr.

Yu, H.; Edunov, S.; Tian, Y.; and Morcos, A. S. 2020. Playing the lottery with rewards and multiple languages: lottery tickets in RL and NLP. *ArXi v* abs/1906.02768.

Zhang, H.; Goodfellow, I.; Metaxas, D.; and Odena, A. 2018. Self-Attention Generative Adversarial Networks.

Zhou, H.; Lan, J.; Liu, R.; and Yosinski, J. 2019. Deconstructing Lottery Tickets: Zeros, Signs, and the Supermask.