Structural Pruning for Diffusion Models

Gongfan Fang  Xinyin Ma  Xinchao Wang*
National University of Singapore
gongfan@u.nus.edu, maxinyin@u.nus.edu, xinchao@nus.edu.sg

Abstract

Generative modeling has recently undergone remarkable advancements, primarily propelled by the transformative implications of Diffusion Probabilistic Models (DPMs). The impressive capability of these models, however, often entails significant computational overhead during both training and inference. To tackle this challenge, we present Diff-Pruning, an efficient compression method tailored for learning lightweight diffusion models from pre-existing ones, without the need for extensive re-training. The essence of Diff-Pruning is encapsulated in a Taylor expansion over pruned timesteps, a process that disregards non-contributory diffusion steps and ensembles informative gradients to identify important weights. Our empirical assessment, undertaken across several datasets highlights two primary benefits of our proposed method: 1) Efficiency: it enables approximately a 50% reduction in FLOPs at a mere 10% to 20% of the original training expenditure; 2) Consistency: the pruned diffusion models inherently preserve generative behavior congruent with their pre-trained models. Code is available at https://github.com/VainF/Diff-Pruning.

1 Introduction

Generative modeling has undergone significant advancements in the past few years, largely propelled by the advent of Diffusion Probabilistic Models (DPMs) [18, 41, 37]. These models have derived numerous applications ranging from text-to-image generation [40], image editing [58], image translation [45], and even discriminative tasks [2, 1]. The incredible power of DPMs, however, often comes at the expense of considerable computational overhead during both training [49] and inference [43]. This trade-off between performance and efficiency presents a critical challenge in the broader application of these models, particularly in resource-constrained environments.

In the literature, huge efforts have been made to improve diffusion models, which primarily revolved around three broad themes: improving model architectures [41, 39, 52], optimizing training methods [49, 11] and accelerating sampling [46, 43, 12]. As a result, a multitude of well-trained diffusion models has been created in these valuable works, showcasing their potential for various applications [48]. However, the notable challenge still remains: the absence of a general compression method that enables the efficient reuse and customization of these pre-existing models without heavy re-training. Overcoming this gap is of paramount importance to fully harness the power of pre-trained diffusion models and facilitate their widespread application across different domains and tasks.

In this work, we demonstrate the remarkable effectiveness of structural pruning [23, 8, 26, 4] as a method for compressing diffusion models, which offers a flexible trade-off between efficiency and quality. Structural pruning is a classic technique that effectively reduces model sizes by eliminating redundant parameters and sub-structures from networks. While it has been extensively studied in discriminative tasks such as classification [16], detection [54], and segmentation [13], applying structural pruning techniques to Diffusion Probabilistic Models poses unique challenges that necessitate

*Corresponding author

a rethinking of traditional pruning strategies. For example, the iterative nature of the generative process in DPMs, the models’ sensitivity to small perturbations in different timesteps, and the intricate interplay in the diffusion process collectively create a landscape where conventional pruning strategies often fall short.

To this end, we introduce a novel approach called **Diff-Pruning**, explicitly tailored for the compression of diffusion models. Our method is motivated by the observation in previous works [41, 52] that different stages in the diffusion process contribute variably to the generated samples. At the heart of our method lies a Taylor expansion over pruned timesteps, which deftly balances the image content, details, and the negative impact of noisy diffusion steps during pruning. Initially, we show that the objective of diffusion models at late timesteps ($t \to T$) prioritize the high-level content of the generated images during pruning, while the early ones ($t \to 0$) refine the images with finer details. However, it is also observed that, when using Taylor expansion for pruning, the noisy stages with large $t$ cannot provide informative gradients for importance estimation and can even harm the compressed performance. Therefore, we propose to model the trade-off between contents, details, and noises as a pruning problem of the diffusion timesteps, which leads to an efficient and flexible pruning algorithm for diffusion models.

Through extensive empirical evaluations across diverse datasets, we demonstrate that our method achieves substantial compression rates while preserving and in some cases even improving the generative quality of the models. Our experiments also highlight two significant features of Diff-Pruning: efficiency and consistency. For example, when applying our method to an off-the-shelf diffusion model pre-trained on LSUN Church [57], we achieve an impressive compression rate of 50% FLOPs, with only 10% of the training cost required by the original models, equating to 0.5 million steps compared to the 4.4 million steps of the pre-existing models. Furthermore, we have thoroughly assessed the generative behavior of the compressed models both qualitatively and quantitatively. Our evaluations demonstrate that the compressed model can effectively preserve a similar generation behavior as the pre-trained model, meaning that when provided with the same inputs, both models yield consistent outputs. Such consistency further reveals the practicality and reliability of Diff-Pruning as a compression method for diffusion models.

In summary, this paper introduces Diff-Pruning as an efficient method for compressing Diffusion Probabilistic Models, which is able to achieve compression with only 10% to 20% of the training costs compared to pre-training. This work may serve as an initial baseline and provide a foundation for future research aiming to enhance the quality and consistency of compressed diffusion models.

## 2 Related Works

### Efficient Diffusion Models

The existing methodologies principally address the efficiency issues associated with diffusion models via three primary strategies: the refinement of network architectures [41, 52, 37], the enhancement of training procedures [11, 49], and the acceleration of sampling [18, 27, 12]. Diffusion models typically employ U-Net models as denoisers, of which the efficiency can be improved via the introduction of hierarchical designs [40] or by executing the training within a novel latent space [41, 19, 25]. Recent studies also suggest integrating more efficient layers or structures into the denoiser to bolster the performance of the U-Net model [52, 39], thereby facilitating superior image quality learning during the training phase. Moreover, a considerable number of studies concentrate on amplifying the training efficiency of diffusion models, with some demonstrating that the diffusion training can be expedited by modulating the weights allocated to distinct timesteps [43, 11]. The training efficiency can also be advanced by learning diffusion models at the patch level [49]. In addition, some approaches underscore the efficiency of sampling, which typically does not necessitate the retraining of diffusion models [27]. In this area, numerous studies aim to diminish the required steps through methods such as early stopping [34] or distillation [43].

### Network Pruning

In recent years, the field of network acceleration [59, 3, 20, 53, 51, 29, 30] has seen notable progress through the deployment of network pruning techniques [51, 16, 33, 23, 14, 5, 15]. The taxonomy of pruning methodologies typically bifurcates into two main categories: structural pruning [23, 6, 56, 26, 50] and unstructured pruning [38, 7, 44, 22]. The distinguishing trait of structural pruning is its ability to physically eliminate parameters and substructures from networks, while unstructured pruning essentially masks parameters by zeroing them out [6, 4]. However, the preponderance of network pruning research is primarily focused on discriminative
tasks, particularly classification tasks [16]. A limited number of studies have ventured into examining the effectiveness of pruning in generative tasks, such as GAN compression [24, 47]. Moreover, the application of structural pruning techniques to Diffusion Probabilistic Models introduces unique challenges that demand a reevaluation of conventional pruning strategies. In this work, we introduce the first dedicated method explicitly designed for pruning diffusion models, which may serve as a useful baseline for future works.

3 Diffusion Model Objectives

Given a data distribution \( q(x) \), diffusion models aim to model a generative distribution \( p_\theta(x) \) to approximate \( q(x) \), taking the form

\[
p_\theta(x) = \int p_\theta(x_{0:T})d x_{1:T}, \quad \text{where} \quad p_\theta(x_{0:T}) := p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)
\]  

(1)

And \( x_1, ..., x_T \) refer to the latent variables, which contribute to the joint distribution \( p_\theta(x_{0:T}) \) with learned Gaussian transitions \( p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \). Diffusion Models involve two opposite processes: a forward (diffusion) process \( q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t} x_{t-1}, \beta_t I) \) that adds noises to the \( x_{t-1} \), based on a pre-defined variance schedule \( \beta_{1:T} \); and a reverse process \( q(x_{t-1}|x_t) \) which “denoises” the observation \( x_t \) to get \( x_{t-1} \). Using the notation \( \alpha_t = 1 - \beta_t \) and \( \bar{\alpha}_t = \prod_{i=1}^{t} \alpha_s \), DDPMs [18] trains a noise predictor with the objective:

\[
L(\theta) := \mathbb{E}_{t, x_0 \sim q(x), \epsilon \sim \mathcal{N}(0, I)} \left[ \| \epsilon - \epsilon_\theta(\sqrt{\alpha_t} x_0 + \sqrt{1-\alpha_t} \epsilon, t) \|^2 \right]
\]

(2)

where \( \epsilon \) is a random noise drawn from a fixed Gaussian distribution and \( \epsilon_\theta \) refers to a learned noise predictor, which is usually an U-Net autoencoder [42] in practice. After training, synthetic images \( x_0 \) can be sampled through an iterative process from a noise \( x_T \sim \mathcal{N}(0, I) \) with the formular:

\[
x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left( x_t - \frac{\beta_t}{\sqrt{1-\alpha_t}} \epsilon_\theta(x_t, t) \right) + \sigma_t z
\]

(3)

where \( z \sim \mathcal{N}(0, I) \) for steps \( t > 1 \) and \( z = 0 \) for \( t = 1 \). In this work, we aim to craft a lightweight \( \epsilon_{\theta'} \) by removing redundant parameters of \( \epsilon_\theta \), which are expected to produce similar \( x_0 \) while the same \( x_T \) are presented.

4 Structural Pruning for Diffusion Models

Given the parameter \( \theta \) of a pre-trained diffusion model, our goal is to craft a lightweight \( \theta' \) by removing sub-structures from the network following existing paradigms [33, 8]. Without loss of generality, we assume that the parameter \( \theta \) is a simple 2-D matrix, where each sub-structure \( \theta_i = [\theta_{i0}, \theta_{i1}, ..., \theta_{iK}] \) is a row vector that contains \( K \) scalar parameters. Structural pruning aims to find a sparse parameter matrix \( \theta' \) that maximally preserves the original performance. Thus, a natural choice is to optimize the loss disruption caused by pruning:

\[
\min_{\theta'} |L(\theta') - L(\theta)|, \quad \text{s.t.} \quad \|\theta'\|_0 \leq s
\]

(4)

The term \( \|\theta'\|_0 \) denotes the L-0 norm of the parameters, which counts the number of non-zero row vectors, and \( s \) represents the sparsity of the pruned model. Nevertheless, due to the iterative nature intrinsic to diffusion models, the training objective, denoted by \( L \), can be perceived as a composition of \( T \) interconnected tasks: \( \{L_1, L_2, ..., L_T\} \). Each task affects and depends on the others, thereby posing a new challenge distinct from traditional pruning problems, which primarily concentrate on optimizing a single objective. In light of the pruning objective as defined in Equation (4) we initially delve into the individual contributions of each loss component, \( L_t \) in pruning, and subsequently propose a tailored method, Diff-Pruning, designed for diffusion models pruning.

Taylor Expansion at \( L_t \)  Initially, we need to model the contribution of \( L_t \) for structural pruning. This work leverages Taylor expansion [33] on \( L_t \) to linearly approximate the loss disruption:

\[
L_t(\theta') = L_t(\theta) + \nabla L_t(\theta)(\theta' - \theta) + O(\|\theta' - \theta\|^2)
\]

\[
\Rightarrow L_t(\theta') - L_t(\theta) = \nabla L_t(\theta)(\theta' - \theta) + O(\|\theta' - \theta\|^2)
\]

(5)
Pruned Noisy Steps

This error has a direct impact on the subsequent input, given by $L_{t_0} = [\theta \cdot \nabla \theta \cdot L_{t_0}]$.

The generated images produced by 5%-pruned DDPMs (without post-training) are illustrated.

### Pruning Equations

The standard Taylor expansion for multiple variables, as described in the literature [9], advocates using $\sum_k \theta_{ik} \cdot \nabla_{\theta_{ik}} \mathcal{L}(\theta, x)$ for importance estimation. This method exclusively takes into account the loss difference between the initial state $\theta$ and the final states $\theta'$. However, considering the iterative nature of diffusion models, even minor fluctuations in loss can influence the final generation results. To this end, we propose to aggregate the influence of removing each parameter as the final importance. This modification models cumulative loss disturbance induced by each $\theta_{ik}$’s removal and leads to a slightly different score function for structural pruning:

$$I_{t}(\theta, x) = |L_{t}(\theta_{ik=0}) - L_{t}(\theta)| = |(\theta_{ik} - \theta_{ik}) \cdot \nabla_{\theta_{ik}} + \cdots + (\theta_{ik} - \theta_{ik}) \cdot \nabla_{\theta_{ik}}|$$

$$= |\theta_{ik} \cdot \nabla_{\theta_{ik}} \mathcal{L}(\theta, x)|$$

where $\nabla_{\theta_{ik}}$ refer to $\nabla_{\theta_{ik}} \mathcal{L}(\theta, x)$. In structural pruning, we aim to remove the entire vector $\theta’_i$ concurrently. The standard Taylor expansion for multiple variables, as described in the literature [9], advocates using $\sum_k \theta_{ik} \cdot \nabla_{\theta_{ik}} \mathcal{L}(\theta, x)$ for importance estimation. This method exclusively takes into account the loss difference between the initial state $\theta$ and the final states $\theta’$. However, considering the iterative nature of diffusion models, even minor fluctuations in loss can influence the final generation results. To this end, we propose to aggregate the influence of removing each parameter as the final importance. This modification models cumulative loss disturbance induced by each $\theta_{ik}$’s removal and leads to a slightly different score function for structural pruning:

$$I_{t}(\theta, x) = \sum_k |L_{t}(\theta_{ik=0}) - L_{t}(\theta)| = \sum_k |\theta_{ik} \cdot \nabla_{\theta_{ik}} \mathcal{L}(\theta, x)|$$

In the following sections, we utilize Equation 7 as the importance function to identify non-critical parameters in diffusion models.

### The Contribution of $L_{t}$

With the Taylor expansion framework, we further explore the contribution of different loss terms $\{L_{1}, ..., L_{T}\}$ during pruning. We consider the functional error $\delta_t = \epsilon_{\theta'}(x, t) - \epsilon_{\theta}(x, t)$ which represents the prediction error for the same inputs at time step $t$. The reverse process allows us to examine the effects $\delta_t \rightarrow 0$ on the generated images $x_0$ by iteratively applying the Equation 3 starting from $\epsilon_{\theta'}(x, t) = \epsilon_{\theta}(x, t) + \delta_t$. At the $t − 1$ step, it leads to the error $\delta_{t-1}$ derived as:

$$\delta_{t-1} = \left[ \frac{1}{\sqrt{\alpha_t}} \left( x_t - \frac{\beta_t}{\sqrt{1 - \alpha_t}} \epsilon_{\theta}(x_{t}, t) \right) + \sigma_t z \right] - \left[ \frac{1}{\sqrt{\alpha_t}} \left( x_t - \frac{\beta_t}{\sqrt{1 - \alpha_t}} (\epsilon_{\theta}(x_{t}, t) + \delta_t) \right) + \sigma_t z \right]$$

$$= \frac{1}{\sqrt{\alpha_t}} \frac{\beta_t}{\sqrt{1 - \alpha_t}} \delta_t$$

This error has a direct impact on the subsequent input, given by $x'_{t-1} = x_{t-1} + \delta_{t-1}$. By checking Equation 4, we can observe that these perturbed inputs can further trigger a chained effect through both $\frac{1}{\sqrt{\alpha_t - 1}} x'_{t-1}$ and $\frac{\beta_t - 1}{\sqrt{1 - \alpha_t - 1}} \epsilon_{\theta'}(x'_{t-1}, t - 1)$. In the first term, the distortion progressively amplifies by a factor $\frac{1}{\sqrt{\alpha_t - 1}} > 1$, which means that this error will be enhanced throughout the generation process. Regarding the second term, pruning affects both the functionality parameterized by $\theta'$ and the inputs $x'_{t-1}$, which contributes to the final results in a nonlinear and more complicated manner, resulting in a more substantial disturbance on the generated images.
As a result, prediction errors occurring at larger $t$ tend to have a larger impact on the images due to the chain effect, which might change the global content of generated images. Conversely, smaller $t$ values focus on refining the images with relatively small modifications. These findings align with our empirical examination using Taylor expansion as illustrated in Figure 1 as well as the observation in previous works [18, 52], which shows that diffusion models tend to generate object-level information at larger $t$ values and fine-tune the features at smaller ones. To this end, we model the pruning problem as a weighted trade-off between contents and details by introducing $\alpha_t$, which acts as a weighting variable for different timesteps $t$. Nevertheless, unconstrained reweighting can be highly inefficient, as it entails exploring a large parameter space for $\alpha_t$ and requires at least $T$ forward-backward passes for Taylor expansion. This results in a vast sampling space and can lead to inaccuracies in the linear approximation. To address this issue, we simplify the re-weighting strategy by treating it as a “pruning problem”, where $\alpha_t$ takes the value of either 0 or 1 for all steps, allowing us to only leverage partial steps for pruning. The general importance metric is modeled as the following.

$$I(\theta_t, x) = \sum_k |\theta_{ik}| \cdot \sum_{s=0}^{t} \nabla_{\theta_{ik}} \mathcal{L}_t(\theta, x)$$ \hspace{1cm} \text{s.t.} \hspace{1cm} \alpha_t \in \{0, 1\} \hspace{1cm} (9)$$

**Taylor Score over Pruned Timesteps.** In Equation 9 we try to remove some “unimportant” timesteps in the diffusion process so as to enable an efficient and stable approximation for partial steps. Our empirical results, as will be discussed in the experiments, indicate two key findings. Firstly, we note that the timesteps responsible for generating content are not exclusively found towards the end of the diffusion process ($t \to T$). Instead, there are numerous noisy and redundant timesteps that contribute minorly to the overall generation, which is similar to the observations in the related work [54]. Secondly, we discovered that employing the full-step objective can sometimes yield suboptimal results compared to using a partial objective. We attribute this negative impact to the presence of converged gradients in the noisy steps ($t \to T$). Taylor approximation in Equation 5 comprises both first-order gradients and higher-order terms. When the loss $\mathcal{L}_t$ converges, the loss curve is predominantly influenced by the higher-order terms rather than the first-order gradients we utilize. Our experiments on several datasets and diffusion models show that the loss term $\mathcal{L}_t$ rapidly approaches 0 as $t \to T$. For example in Figure 5 the relative loss $\frac{\mathcal{L}_t}{\mathcal{L}_{max}}$ of a pre-trained diffusion model for CIFAR-10 decreases to 0.05 when $t = 250$. Consequently, a full Taylor expansion can accumulate a considerable amount of noisy gradients from these converged or unimportant steps, resulting in an inaccurate estimation of weight importance.
Considering the significant impact of larger timesteps, it is necessary to incorporate them for importance estimation. To address this problem, Equation 9 naturally provides a simple and practical thresholding strategy for pruning. To achieve this, we introduce a threshold parameter $T$ based on the relative loss $\frac{L_t}{L_{\text{max}}}$. Those timesteps with a relative loss below this threshold, i.e., $\frac{L_t}{L_{\text{max}}} < T$, are considered uninformative and are disregarded by setting $\alpha_t = 0$, which yields the finalized importance score:

$$I(\theta_i, x) = \sum_k \theta_{ik} \cdot \sum_{\{t | \frac{L_t}{L_{\text{max}}} > T\}} \nabla_{\theta_{ik}} L_t(\theta, x)$$

(10)

In practice, we need to select an appropriately large value for $T$ to strike a well-balanced preservation of details and content, while also avoiding uninformative gradients from noisy loss terms. The full algorithm is summarized in Alg. 1.

5 Experiments

5.1 Settings

Datasets and Models The efficacy of Diff-Pruning is empirically validated across six diverse datasets, including CIFAR-10 (32\times32)\cite{21}, CelebA-HQ (64\times64)\cite{32}, LSUN Church (256\times256), LSUN Bedroom (256\times256)\cite{57} and ImageNet-1K (256\times256). We focus on two popular DPMs in our experiments, i.e., Denoising Diffusion Probability Models (DDPMs)\cite{18} and Latent Diffusion Models (LDMs)\cite{41}. For the sake of reproducibility, we utilize off-the-shelf DPMs from\cite{18} and\cite{41} as pre-trained models and prune these models in a one-shot fashion\cite{23}.

Evaluation Metrics In this paper, we concentrate primarily on three types of metrics: 1) Efficiency metrics, which include the number of parameters (#Params) and Multiply-Add Accumulation (MACs); 2) Quality metric, namely the Frechet Inception Distance (FID)\cite{17}; and 3) Consistency metric, represented by Structural Similarity (SSIM)\cite{50}. Unlike previous generative tasks that lacked reference images, we employ the SSIM index to evaluate the similarity between images generated by pre-trained models and pruned models, given identical noise inputs. We deploy a 250-step DDIM sampler\cite{46} for ImageNet and a 100-step DDIM sampler for other experiments.

5.2 An Simple Benchmark for Diffusion Pruning

Scratch Training v.s. Pruning. Table 1 shows our results on CIFAR-10 and CelebA-HQ. The first baseline method that piques our interest is scratch training. Numerous studies on network pruning\cite{10} suggest that training a compact network from scratch can be a formidable competitor. To ensure a fair comparison, we create randomly initialized networks with the same architecture as the pruned ones for scratch training. Our results reveal that scratch training demands relatively more steps to reach convergence. This suggests that training lightweight models from scratch may not be an efficient and economical approach, given its training cost is comparable to that of pre-trained models. Conversely, we observe that all pruning methods are able to converge within approximately 100K steps and outperform scratch training in terms of FID and SSIM scores. Thus, pruning emerges as a potent technique for compressing pre-trained Diffusion Models.

Pruning Criteria. A significant aspect of network pruning is the formulation of pruning criteria, which serve to identify superfluous parameters within networks. Due to the absence of dedicated work on Diffusion model pruning, we adapted three basic pruning methods from discriminative tasks: random pruning, magnitude-based pruning\cite{16}, and Taylor-based pruning\cite{36}, which we refer to as Random, Magnitude, and Taylor respectively in subsequent sections. For a given parameter $\theta$, Random assigns importance scores derived from a uniform distribution to each $\theta_i$ randomly, denoted as $I(\theta) \sim U(0, 1)$. This results in a straightforward baseline devoid of any prior or bias, and has been shown to be a competitive baseline for pruning\cite{28}. Magnitude subscribes to the “smaller-norm-less-informative” hypothesis\cite{23}\cite{55}, modelling the weight importance as $I(\theta) = |\theta|$. In contrast, Taylor is a data-driven criterion that measures importance as $I(\theta, x) = |\theta \cdot \nabla_{\theta} L(x, \theta)|$, which aims to minimize loss change as discussed in our method. As shown in\cite{1} an intriguing phenomenon is that these three baseline methods do not maintain a consistent ranking on these two datasets. For
<table>
<thead>
<tr>
<th>Method</th>
<th>CIFAR-10 32 × 32 (100 DDIM steps)</th>
<th>CelebA-HQ 64 × 64 (100 DDIM steps)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#Params</td>
<td>MACs</td>
</tr>
<tr>
<td>Pretrained</td>
<td>35.7M</td>
<td>6.1G</td>
</tr>
<tr>
<td>Scratch Training</td>
<td>19.8M</td>
<td>3.4G</td>
</tr>
<tr>
<td>Scratch Training</td>
<td>5.68</td>
<td>0.905</td>
</tr>
<tr>
<td>Random Pruning</td>
<td>5.62</td>
<td>0.926</td>
</tr>
<tr>
<td>Magnitude Pruning</td>
<td>5.48</td>
<td>0.929</td>
</tr>
<tr>
<td>Taylor Pruning</td>
<td>5.56</td>
<td>0.928</td>
</tr>
<tr>
<td>Ours (T = 0.00)</td>
<td>5.49</td>
<td>0.932</td>
</tr>
<tr>
<td>Ours (T = 0.02)</td>
<td>5.44</td>
<td>0.931</td>
</tr>
<tr>
<td>Ours (T = 0.05)</td>
<td>5.44</td>
<td>0.931</td>
</tr>
</tbody>
</table>

Table 1: Diffusion pruning on CIFAR-10 and CelebA. We leverage Frechet Inception Distance (FID) and Structural Similarity (SSIM) to estimate the quality and similarity of generated samples under the same random seed. A larger SSIM score means more consistent generation.

instance, while Magnitude achieves the best FID performance among the three on CIFAR-10, it performs poorly on CelebA datasets. In contrast, our method delivers stable improvements over baseline methods, demonstrating superior performance on both datasets. Remarkably, our method even surpasses the pre-trained model on CelebA-HQ, with only 100K optimizations. Nonetheless, performance degradation is observed on CIFAR-10, which can be attributed to its more complex scene and a larger number of categories.

5.3 Pruning at Higher Resolutions

DDPMs on LSUN  To further validate the efficiency and effectiveness of our proposed Diff-Pruning, we perform pruning experiments on two 256×256 scene datasets, LSUN Church, and LSUN Bedroom. The pre-trained models from require around 2.4M and 4.4M training steps, which can be quite time-consuming in practice. We demonstrate that Diff-Pruning can compress these pre-existing models using only 10% of the standard training resources. We report the number of parameters, MACs, and FID scores in Table 2, and compare the pruned methods to the pre-trained ones as well as those trained from scratch. Results show that the pruned model converges with a passable FID score in 10% of the standard steps, while a model trained from scratch is still severely under-fitted. Nevertheless, we also discover that compressing a model trained on large-scale datasets, such as LSUN Bedroom, which contains 300K images, proves to be quite challenging with a very limited number of training steps. We show that, in the supplementary materials, the FID scores can be further improved with more training steps. Moreover, we also visualize the generated images in Figure 2, and report the single-image SSIM score to measure the similarity of generated images. By nature, the pruned models can preserve similar generation capabilities since they inherit most parameters from the pre-trained models.

Conditional LDMs on ImageNet  Table 3 and Figure 3 illustrate the pruning results of LDM pre-trained on ImageNet-1K. An LDM consists of an encoder, a decoder, and a U-Net model. Around
Figure 2: Generated images of the pre-trained models [18] (left) and the pruned models (right) on LSUN Church and LSUN Bedroom. SSIM measures the similarity between generated images.

Table 2: Pruning diffusion models on LSUN Church and LSUN Bedroom.

<table>
<thead>
<tr>
<th>Method</th>
<th>#Params</th>
<th>MACs</th>
<th>FID</th>
<th>Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pretrained</td>
<td>113.7M</td>
<td>248.7G</td>
<td>10.6</td>
<td>4.4M</td>
</tr>
<tr>
<td>Scratch Training</td>
<td>63.2M</td>
<td>138.8G</td>
<td>40.2</td>
<td>0.5M</td>
</tr>
<tr>
<td>Ours ($T = 0.01$)</td>
<td>63.2M</td>
<td>138.8G</td>
<td><strong>13.9</strong></td>
<td>0.5M</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>#Params</th>
<th>MACs</th>
<th>FID</th>
<th>Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pretrained</td>
<td>113.7M</td>
<td>248.7G</td>
<td>6.9</td>
<td>2.4M</td>
</tr>
<tr>
<td>Scratch Training</td>
<td>63.2M</td>
<td>138.8G</td>
<td>50.3</td>
<td>0.2M</td>
</tr>
<tr>
<td>Ours ($T = 0.01$)</td>
<td>63.2M</td>
<td>138.8G</td>
<td><strong>18.6</strong></td>
<td>0.2M</td>
</tr>
</tbody>
</table>

Table 2: Pruning diffusion models on LSUN Church and LSUN Bedroom.

400M parameters come from the U-Net architecture and only 55M from the autoencoder. Therefore, we mainly focus on the pruning of the U-Net model. We used the threshold $T = 0.1$ to ignore those converged layers and make the pruning process more efficient. With $T = 0.1$, only 534 steps participate in the pruning process. After importance estimation, we apply a pre-defined channel sparsity of 30% to all layers, leading to a lightweight U-Net with 189.43M parameters. Finally, we finetune the pruned model for only 4 epochs with the official training script, with a scaled learning rate of $0.1 \times \text{lr}_{\text{base}}$.

5.4 Ablation Study

Pruned Timesteps. First, we conduct an empirical study evaluating the partial Taylor expansion over pruned timesteps. This approach prioritizes steps with larger gradients and strives to preserve as much content and detail as possible, thereby enabling more accurate and efficient pruning. The impacts of timestep pruning are demonstrated in Figure 5. We seek to prune a pre-trained diffusion model over a range of steps, spanning from 50 to 1000, after which we utilize the SSIM metric to gauge the distortion induced by pruning. In diffusion models, earlier steps ($t \rightarrow 0$) usually present larger gradients compared to the later ones ($t \rightarrow T$) [41]. This inherently leads to gradients that have reached a convergence when $t$ is large. In the CIFAR-10 dataset, we find that the optimal SSIM score...
can be attained at around 250 steps, and adding more steps can slightly deteriorate the quality of the synthetic images. This primarily stems from the inaccuracy of the first-order Taylor expansion at converged points, where the gradient no longer provides useful information and can even distort informative gradients through accumulation. However, we observe that the situation differs slightly with the CelebA dataset, where more steps can be beneficial for importance estimation.

**Pruning Ratios.** Table 4 presents the #Params, MACs, FID, and SSIM scores of models subjected to various pruning ratios based on MACs. Notably, our findings reveal that, unlike CNNs employed in discriminative models, diffusion models exhibit a significant sensitivity to changes in model size. Even a modest pruning ratio of 16% leads to a noticeable degradation in FID score (4.19 → 4.62). In classification tasks, a perturbation in loss does not necessarily impact the final accuracy; it may only undermine prediction confidence while leaving classification accuracy unaffected. However, in generative models, the FID score is very sensitive, making it more susceptible to domain shift.

**Thresholding.** In addition, we conducted experiments to investigate the impact of the thresholding parameter $T$. Setting $T = 0$ corresponds to a full Taylor expansion at all steps, while $T > 0$ denotes pruning of certain timesteps during importance estimation. The quantitative findings presented in Table 5 align with the SSIM results depicted in Figure 5. Notably, Diff-Pruning attains optimal performance when the quality of generated images reaches its peak. For datasets such as CIFAR-10, we observed that a 200-step Taylor expansion is sufficient to achieve satisfactory results. Besides, using a full Taylor expansion, in this case, can be detrimental, as it accumulates noisy gradients over approximately 700 steps, which obscures the correct gradient information from earlier steps.

---

**Figure 3:** Images sampled from the pruned conditional LDM on ImageNet-1K-256

<table>
<thead>
<tr>
<th>Method</th>
<th>#Params</th>
<th>MACs</th>
<th>FID</th>
<th>IS</th>
<th>Train Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pretrained LDM</td>
<td>400.92M</td>
<td>99.80G</td>
<td>3.60</td>
<td>247.67</td>
<td>2000K</td>
</tr>
<tr>
<td>Scratch Training</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Taylor Pruning</td>
<td>189.43M</td>
<td>52.71G</td>
<td>11.18</td>
<td>138.97</td>
<td>100K</td>
</tr>
<tr>
<td>Ours ($T = 0.1$)</td>
<td>9.16</td>
<td>201.81</td>
<td>9.16</td>
<td>201.81</td>
<td>100K</td>
</tr>
</tbody>
</table>

Table 3: Compressing conditional Latent Diffusion Models on ImageNet-1K (256 × 256)

<table>
<thead>
<tr>
<th>Pruning Ratios</th>
<th>#Params</th>
<th>MACs</th>
<th>FID</th>
<th>SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>35.7M</td>
<td>6.1G</td>
<td>4.19</td>
<td>1.000</td>
</tr>
<tr>
<td>16%</td>
<td>27.5M</td>
<td>5.1G</td>
<td>4.62</td>
<td>0.942</td>
</tr>
<tr>
<td>44%</td>
<td>19.8M</td>
<td>3.4G</td>
<td>5.29</td>
<td>0.932</td>
</tr>
<tr>
<td>56%</td>
<td>14.3M</td>
<td>2.7G</td>
<td>6.36</td>
<td>0.922</td>
</tr>
<tr>
<td>70%</td>
<td>8.6M</td>
<td>1.5G</td>
<td>9.33</td>
<td>0.909</td>
</tr>
</tbody>
</table>

Table 4: Pruning with different ratios

<table>
<thead>
<tr>
<th>Thresholding</th>
<th>Threshold</th>
<th>Steps</th>
<th>FID</th>
<th>SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T = 0.00$</td>
<td>1000</td>
<td>5.49</td>
<td>0.932</td>
<td></td>
</tr>
<tr>
<td>$T = 0.01$</td>
<td>707</td>
<td>5.41</td>
<td>0.932</td>
<td></td>
</tr>
<tr>
<td>$T = 0.02$</td>
<td>433</td>
<td>5.44</td>
<td>0.931</td>
<td></td>
</tr>
<tr>
<td>$T = 0.05$</td>
<td>244</td>
<td>5.29</td>
<td>0.932</td>
<td></td>
</tr>
<tr>
<td>$T = 0.10$</td>
<td>127</td>
<td>5.31</td>
<td>0.931</td>
<td></td>
</tr>
</tbody>
</table>

Table 5: Pruning with different threshold $T$
Visualization of Different Importance Criteria. Figure 4 visualizes the images generated by pruned models using different pruning criteria, including the proposed method with $T = 0$ (w/o timestep pruning) and $T > 0$. The SSIM scores of the generated samples are reported for a quantitative comparison. The Diff-Pruning method with $T > 0$ achieves superior visual quality, with an SSIM score of 0.905 after pruning. It is observed that employing more timesteps in our method could have a negative impact, leading to greater distortion in both textures and contents.

6 Conclusion

This work introduces Diff-Pruning, a dedicated method for compressing diffusion models. It utilizes Taylor expansion over pruned timesteps to identify and remove non-critical parameters. The proposed approach is capable of crafting lightweight yet consistent models from pre-trained ones, incurring only about 10% to 20% of the cost compared to pre-training. This work may set an initial baseline for future research that aims at improving both the generation quality and the consistency of pruned diffusion models.

Acknowledgment

This research is supported by the Ministry of Education, Singapore, under its Academic Research Fund Tier 2 (Award Number: MOE-T2EP20122-0006).
References


