Attention-Driven Training-Free Efficiency Enhancement of Diffusion Models

Hongjie Wang[1]*, Difan Liu[2], Yan Kang[2], Yijun Li[2], Zhe Lin[2], Niraj K. Jha[1], Yuchen Liu[2]†
[1]Princeton University, [2]Adobe Research

Abstract

*Diffusion models (DMs) have exhibited superior performance in generating high-quality and diverse images. However, this exceptional performance comes at the cost of expensive generation process, particularly due to the heavily used attention module in leading models. Existing works mainly adopt a retraining process to enhance DM efficiency. This is computationally expensive and not very scalable. To this end, we introduce the **A**ttention-driven **T**raining-free **E**fficient **D**iffusion **M**odel (**AT-EDM**) framework that leverages attention maps to perform run-time pruning of redundant tokens, without the need for any retraining. Specifically, for single-denoising-step pruning, we develop a novel ranking algorithm, Generalized Weighted Page Rank (G-WPR), to identify redundant tokens, and a similarity-based recovery method to restore tokens for the convolution operation. In addition, we propose a Denoising-Steps-Aware Pruning (DSAP) approach to adjust the pruning budget across different denoising timesteps for better generation quality. Extensive evaluations show that AT-EDM performs favorably against prior art in terms of efficiency (e.g., 38.8% FLOPs saving and up to 1.53× speed-up over Stable Diffusion XL) while maintaining nearly the same FID and CLIP scores as the full model. Project webpage: https://atedm.github.io.*

1. Introduction

Diffusion Models (DMs) [9, 29] have revolutionized computer vision research by achieving state-of-the-art performance in various text-guided content generation tasks, including image generation [28], image editing [12], super resolution [17], 3D objects generation [27], and video generation [10]. Nonetheless, the superior performance of DMs comes at the cost of an enormous computation budget. Although Latent Diffusion Models (LDMs) [28, 34] make text-to-image generation much more practical and affordable for normal users, their inference process is still too slow. For example, on the current flagship mobile phone, generating a single 512px image requires 90 seconds [19].

To address this issue, numerous approaches geared at efficient DMs have been introduced, which can be roughly categorized into two regimes: (1) efficient sampling strategy [24, 30] and (2) efficient model architecture [19, 38]. While efficient sampling methods can reduce the number of denoising steps, they cannot reduce the memory footprint, making it still challenging to use on devices with limited memory. On the contrary, an efficient architecture reduces the cost of each step and can be further combined with sampling strategies to achieve even better efficiency. However, most prior efficient architecture works **require retraining** of the DM backbone, which can take thousands of A100 GPU hours. Moreover, due to different deployment settings on various platforms, different compression ratios of the backbone model are required, which necessitate multiple retraining runs later. Such retraining costs are a big concern even for large companies in the industry.

To this end, we propose the **Attention-driven Training-free Efficient Diffusion Model (AT-EDM)** framework, which accelerates DM inference at run-time without any retraining. To the best of our knowledge, training-free architectural compression of DMs is a highly uncharted area. Only one prior work, Token Merging (ToMe) [1], addresses this problem. While ToMe demonstrates good performance on Vision Transformer (ViT) acceleration [2], its performance on DMs still has room to improve. To further enrich research on training-free DMs, we start our study by profiling the floating-point operations per second (FLOPs) of the state-of-the-art model, Stable Diffusion XL (SD-XL) [26], through which we find that attention blocks are the dominant workload. In a single denoising step, we thus propose to dynamically prune redundant tokens to accelerate attention blocks. We pioneer a fast graph-based algorithm, Generalized Weighted Page Rank (G-WPR), inspired by ZeroTPrune [35], and deploy it on attention maps in DMs to identify superfluous tokens. Since SD-XL contains ResNet blocks, which require a full number of tokens for the convolution operations, we propose a novel similarity-based token copy approach to recover pruned tokens, again leveraging attention maps. This token recovery method is critical to maintaining image quality. We find that naive interpolation or padding of pruned tokens adversely impacts gener-

*Work was partly done during an internship at Adobe.
†Corresponding author.

Figure 1. Examples of applying AT-EDM to SD-XL [26]. Compared to the full-size model (**top row**), our accelerated model (**bottom row**) has around 40% FLOPs reduction while enjoying competitive generation quality at various aspect ratios.

ation quality severely. In addition to single-step architectural pruning, we also investigate cross-step redundancy in the denoising process by analyzing the variance of attention maps. This leads us to a novel pruning schedule, dubbed as Denoising-Steps-Aware Pruning (DSAP) schedule, where we adjust the pruning ratios across different denoising steps. We find DSAP not only significantly improves our method, but also helps improve other run-time pruning methods like ToMe [1]. Compared to ToMe, our approach shows a clear improvement by generating clearer objects with sharper details and better text-image alignment under the same acceleration ratio. In summary, our contributions are four-fold:

- We propose the AT-EDM framework, which leverages rich information from attention maps to accelerate pre-trained DMs without retraining.
- We design a token pruning algorithm for a single denoising step. We pioneer a fast graph-based algorithm, G-WPR, to identify redundant tokens, and a novel similarity-based copy method to recover missing tokens for convolution.
- Inspired by the variance trend of attention maps across denoising steps, we develop the DSAP schedule, which improves generation quality by a clear margin. The schedule also provides improvements over other run-time acceleration approaches, demonstrating its wide applicability.
- We use AT-EDM to accelerate a top-tier DM, SD-XL, and conduct both qualitative and quantitative evaluations. Noticeably, our method shows comparable performance with an FID score of 28.0 with 40% FLOPs reduction relative to the full-size SD-XL (FID 27.3), achieving state-of-the-art results. Visual examples are shown in Fig. 1.

2. Related Work

Text-to-Image Diffusion Models. The diffusion-based generative models enable high-fidelity image synthesis with variant text prompts [4, 9]. However, DMs in the pixel space suffer from large generation latency, which severely limits their applications [36]. LDM [28] encodes the pixel space into a latent space and deploys a DM in the latent space. This reduces computational cost significantly while maintaining generation quality. Subsequently, improved versions of the LDM, called Stable Diffusion Models (SDMs), have been released. The most recent and powerful one is SD-XL [26], which is our default backbone in this work.

Efficient Diffusion Models. Researchers have made enormous efforts to make DMs more efficient. Existing efficient DMs can be divided into two types: (1) **Efficient sampling** to reduce the required number of denoising steps [22, 30–32]. A recent efficient sampling work [24] managed to reduce the number of denoising steps to one by iterative distillation. (2) **Architectural compression** to make each sampling step more efficient [11, 19, 36, 38]. A recent work [13] removes multiple ResNet and attention blocks in the U-Net through distillation. Although these methods can reduce computational costs while maintaining decent image quality, they require *expensive retraining* of the DM backbone to enhance efficiency. Thus, a training-free method to enhance the efficiency of DMs is needed. Note that our proposed training-free framework, AT-EDM, is **orthogonal** to these methods and can be stacked with them to further improve their efficiency. We provide corresponding experimental evidence in Supplementary Material (Supp).

Training-Free Efficiency Enhancement. Training-free (i.e., post-training) efficiency enhancement schemes have been widely explored for CNNs [14, 33, 39] and ViTs [2, 7, 15, 35]. However, training-free schemes for DMs are still poorly explored. To the best of our knowledge, the only prior work in this field is ToMe [1]. It uses token embedding vectors to obtain pair-wise similarity and merges similar tokens to reduce computational overheads. While ToMe achieves a decent speed-up when applied to SD-v1.x and SD-v2.x, we find that it does not help much when ap-

plied to the state-of-the-art DM backbone, SD-XL, whilst our method achieves a clear improvement over it (see Section 4). This is mainly due to (1) the significant architectural change of SD-XL (see Supp); (2) our better algorithm design to identify redundant tokens.

Exploiting Attention Maps. We take inspiration from recent image editing works [3, 5, 8, 25], in which attention maps clearly demonstrate which parts of a generated image are more important. This inspires us to use the correlations and couplings between tokens indicated by attention maps to identify unimportant tokens and prune them. Specifically, we can convert attention maps to directed graphs, where nodes represent tokens, without information loss. Based on this idea, we develop the G-WPR algorithm for token pruning in a single denoising step.

Non-Uniform Denoising Steps. Various existing works [6, 18, 21, 37] demonstrate that denoising steps contribute differently to the quality of generated images; thus, it is not optimum to use uniform denoising steps. OMS-DPM [21] uses different models in different denoising steps. DDSM [37] adapts model size to the importance of each denoising step. AutoDiffusion [18] employs evolutionary search to skip some denoising steps and blocks in the U-Net. Diff-Pruning [6] uses a Taylor expansion over timesteps to disregard non-contributory diffusion steps. All existing methods either require an intensive training/fine-tuning/searching process to obtain and deploy the desired denoising schedule or are not compatible with our proposed G-WPR token pruning algorithm due to the U-Net architecture change. On the contrary, based on our investigation of the variance of attention maps across denoising steps, we propose DSAP. Its schedule can be determined via simple ablation experiments and it is compatible with any token pruning scheme. DSAP can potentially be migrated to existing efficient DMs to help improve their image quality.

3. Methodology

We start our investigation by profiling the FLOPs of the state-of-the-art DM, SD-XL, as shown in Fig. 2. Noticeably, among compositions of the sampling module (U-Net), attention blocks, which consist of several consecutive attention layers, dominate the workload for image generation. Therefore, we propose AT-EDM to accelerate attention blocks in the model through token pruning. AT-EDM contains two important parts: a single-denoising-step token pruning scheme and the DSAP schedule. We provide an overview of these two parts and then discuss them in detail.

3.1. Overview

Fig. 3 illustrates the two main components of AT-EDM:

Part I: Token pruning scheme in a single denoising step. **Step 1:** We can potentially obtain the attention maps from self-attention or cross-attention of an attention layer. We compare the two choices and analyze them in detail through ablation experiments. **Step 2:** We use a scoring module to assign an importance score to each token based on the obtained attention map. We propose an algorithm called G-WPR to assign importance scores to each token (see Section 3.2). **Step 3:** We generate pruning masks based on the calculated importance score distribution. Currently, we simply use the top-k approach to determine the retained tokens, i.e., prune tokens with lower importance scores. **Step 4:** We use the generated mask to perform token pruning. We do this after the feed-forward layer of attention layers. We may also perform pruning early before the feed-forward layers. We provide ablative experimental results for it in Supp. **Step 5:** We repeat Steps 1-4 for consecutive attention layers. Note that we do not apply pruning to the last attention layer before the ResNet layer. **Step 6:** Finally, before passing the pruned feature map to the ResNet block, we need to recover the pruned tokens. We propose a similarity-based copy technique to address this (see Section 3.2).

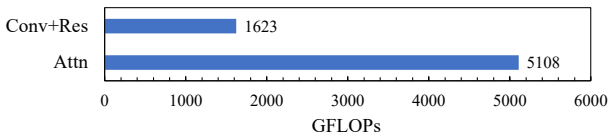

Figure 2. U-Net FLOPs breakdown of SD-XL [26] measured with 1024px image generation. Attention blocks cost the most.

Part II: DSAP schedule. Attention maps in early denoising steps are more chaotic and less informative than those in later steps, which is indicated by their low variance. Thus, they have a weaker ability to differentiate unimportant tokens [8]. Based on this intuition, we design the DSAP schedule that prunes fewer tokens in early denoising steps. Specifically, we select some attention blocks in the up-sampling and down-sampling stages and leave them unpruned, since they contribute more to the generated image quality than other attention blocks [19]. We demonstrate the schedule in detail in Section 3.3.

3.2. Part I: Token Pruning in a Single Step

Notation. Suppose $\mathbf{A}^{(h,l)} \in \mathbb{R}^{M\times N}$ is the attention map of the h-th head in the l-th layer. It reflects the correlations between M Query tokens and N Key tokens. We refer to $\mathbf{A}^{(h,l)}$ as $\mathbf{A}$ for simplicity in the following discussion. Let $A_{i,j}$ denote its element in the i-th row, j-th column. $\mathbf{A}$ can be thought of as the adjacency matrix of a directed graph in the G-WPR algorithm. In this graph, the set of nodes with input (output) edges is referred to as Φ_{in} (Φ_{out}). Nodes in Φ_{in} (Φ_{out}) represent Key (Query) tokens, i.e., $\Phi_{in} = \{k_j\}_{j=1}^{N}$ ($\Phi_{out} = \{q_i\}_{i=1}^{M}$). Let s_K^t (s_Q^t) denote the vector that represents the importance score of Key (Query) tokens in the t-th iteration of the G-WPR algorithm. In the case of self-attention, Query tokens are the same as

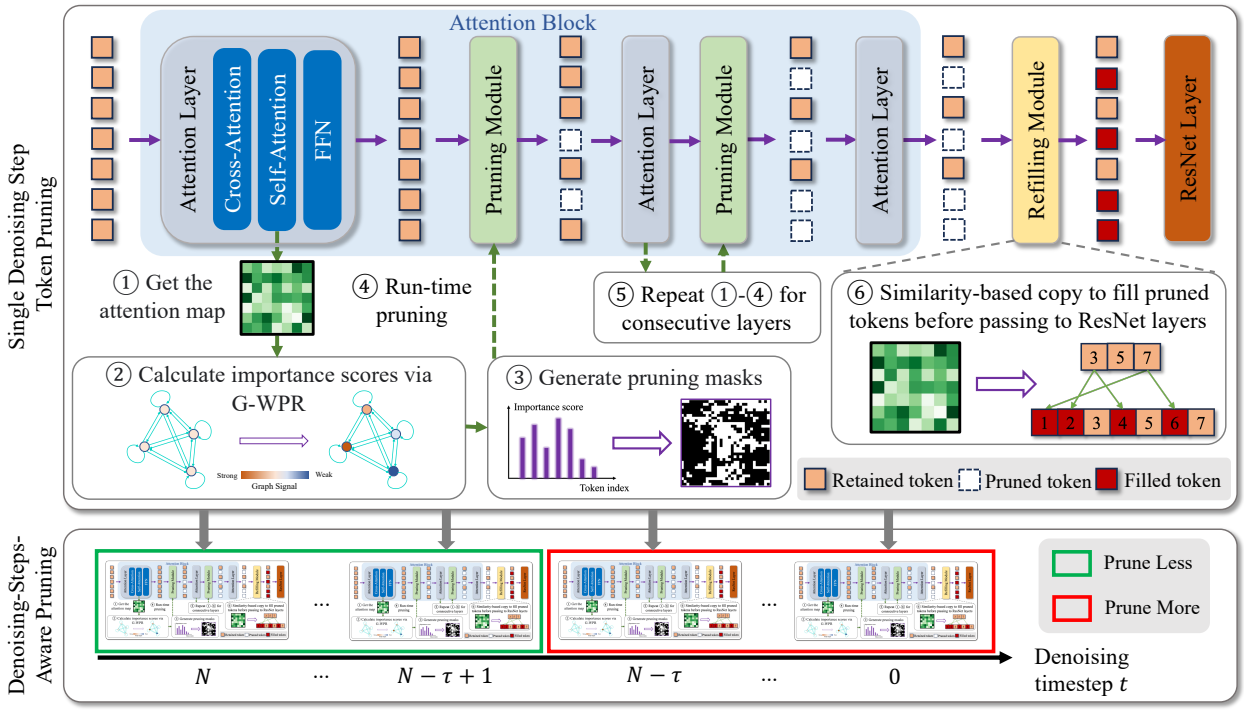

Figure 3. Overview of our proposed framework **AT-EDM**. **Single-Denoising-Step Token Pruning:** (1) we get the attention map from self-attention; (2) we calculate the importance score for each token using G-WPR; (3) we generate pruning masks; (4) we apply the masks to tokens after the feed-forward network to realize token pruning; (5) we repeat Steps (1)-(4) for each consecutive attention layer; (6) we recover pruned tokens through similarity-based copy before the ResNet block. **Denoising-Steps-Aware Pruning Schedule:** In early steps, we propose to prune fewer tokens and to have less FLOPs reduction. In later steps, we prune more aggressively for higher speedup.

Key tokens. Specifically, we let $\{x_i\}_{i=1}^N$ denote the N tokens and s denote their importance scores in the description of our token recovery method.

The G-WPR Algorithm. WPR [35] uses the attention map as an adjacency matrix of a directed complete graph. It uses a graph signal to represent the importance score distribution among nodes in this graph. WPR uses the adjacency matrix as a graph operator, applying it to the graph signal iteratively until convergence. In each iteration, each node votes for which node is more important. The weight of the vote is determined by its importance in the last iteration. However, WPR, as proposed in [35], constrains the used attention map to be a self-attention map. Based on this, we propose the G-WPR algorithm, which is compatible with both self-attention and cross-attention, as shown in Algorithm 1. The attention from Query q_i to Key k_j weights the edge from q_i to k_j in the graph generated by $\mathbf{A}$. In each iteration of the vanilla WPR, by multiplying with the attention map, we map the importance of Query tokens s_Q^t to the importance of Key tokens s_K^{t+1}, i.e., each node in Φ_{out} votes for which Φ_{in} node is more important. For self-attention, $s_Q^{t+1} = s_K^{t+1}$ since Query and Key tokens are the same. For cross-attention, Query tokens are image tokens and Key tokens are text prompt tokens. Based on the intuition that important image tokens should devote a large portion of their attention to important text prompt tokens, we define function $f(\mathbf{A}, s_K)$ that maps s_K^{t+1} to s_Q^{t+1}. One entropy-based implementation is

$$s_Q^{t+1}(q_i) = f(\mathbf{A}, s_K^{t+1}) = \frac{\sum_{j=1}^N A_{i,j} \cdot s_K^{t+1}(k_j)}{-\sum_{j=1}^N A_{i,j} \cdot \ln A_{i,j}} \quad (1)$$

where $A_{i,j}$ is the attention from Query q_i to Key k_j. This is the default setting for cross-attention-based WPR in the following sections. We discuss and compare other implementations in Supp. Note that for self-attention, $f(\mathbf{A}, s_K^{t+1}) = s_K^{t+1}$. The G-WPR algorithm has an $O(M \times N)$ complexity, where M (N) is the number of Query (Key) tokens. We employ this algorithm in each head and obtain the root mean square of scores from different heads (to reward tokens that obtain very high importance scores in a few heads).

Recovering Pruned Tokens. We have fewer tokens after token pruning, leading to efficiency enhancement. However, retained tokens form irregular maps and thus cannot be used for convolution, as shown in Fig. 4. We need to recover the pruned tokens to make them compatible with the following convolutional operations in the ResNet layer. We implement several straightforward token recovery methods as baselines for comparison: (I) Padding Zeros; (II) Interpolation; (III) Direct Copy of input tokens at the locations of pruned tokens (check Supp for details).

To avoid the effect of distribution shift, we propose a **similarity-based copy** technique, as shown in Fig. 4. We select tokens that are similar to pruned tokens from the re-

Algorithm 1 The G-WPR algorithm for both self-attention and cross-attention

Require: $M, N > 0$ is the number of nodes in Φ_{out}, Φ_{in}; $\mathbf{A} \in \mathbb{R}^{M \times N}$; $s_Q \in \mathbb{R}^M, s_K \in \mathbb{R}^N$; $f(\mathbf{A}, s_k)$ maps the importance of Key to that of Query

Ensure: $s \in \mathbb{R}^M$ represents the importance score of image tokens

$s_Q^0 \leftarrow \frac{1}{M} \times e_M$

$t \leftarrow 0$

while $(|s_Q^t - s_Q^{t-1}| > \epsilon)$ **or** $(t = 0)$ **do**

$s_K^{t+1} \leftarrow \mathbf{A}^T \times s_Q^t$

$s_Q^{t+1} \leftarrow f(\mathbf{A}, s_K^{t+1})$

$s_Q^{t+1} \leftarrow s_Q^{t+1} / |s_Q^{t+1}|$

$t \leftarrow t + 1$

end while

$s \leftarrow s_Q^t$

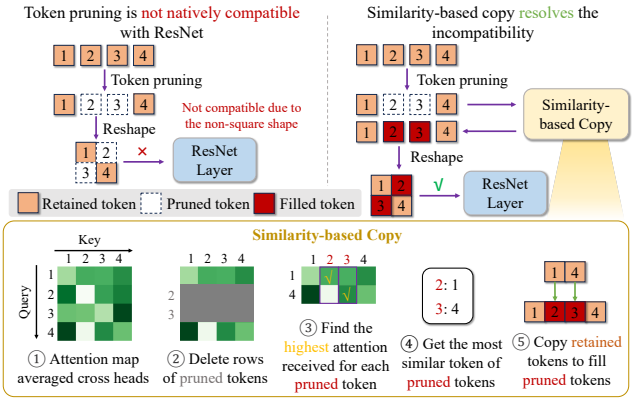

Figure 4. Our similarity-based copy method for token recovering resolves the incompatibility between token pruning and ResNet. Token pruning incurs the non-square shape of feature maps and thus is not compatible with ResNet. To address this, we recover the pruned tokens through their most similar retained tokens. After recovering, tokens can be translated into a spatially-complete feature map to serve as input to ResNet blocks.

tained tokens. We use the self-attention map to determine the source of the highest attention received for each pruned token and consider that as the most similar one. This is based on the intuition that attention from token x_a to token x_b, $A_{a,b}$, is determined by two factors: (1) importance of token x_b, i.e., $s(x_b)$, and (2) similarity between token x_a and x_b. If we observe the attention that x_b receives, i.e., compare $\{A_{i,b}\}_{i \in N}$, since $s(x_b)$ is fixed, index $i = \eta$ that maximizes $\{A_{i,b}\}_{i \in N}$ is the index of the most similar token, i.e., x_η. Finally, we copy the value of token x_η to fill (i.e., recover) the pruned token x_b.

3.3. Part II: Denoising-Steps-Aware Pruning

Early denoising steps determine the layout of generated images and, thus, are crucial. On the contrary, late denoising steps aim at refining the generated image, natively including redundant computations since many regions of the image do not need refinement. In addition, *early denoising steps have a weaker ability to differentiate unimportant tokens*, and late

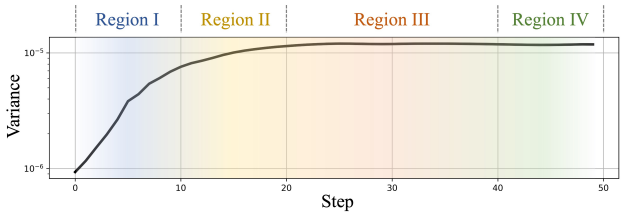

Figure 5. Variance of attention maps in different denoising steps. We divide the denoising steps into four typical regions: (I) Very-early steps: Variance of attention maps is small and increases rapidly; (II) Mid-early steps: Variance of attention maps is large and increases slowly; (III) Middle steps: Variance of attention maps is large and almost constant; (IV) Last several steps.

denoising steps yield informative attention maps and differentiate unimportant tokens better. To support this claim, we investigate the variance of feature maps in different denoising steps, as shown in Fig. 5. It indicates that attention maps in early steps are more uniform. They assign similar attention scores to both important and unimportant tokens, making it harder to precisely identify unimportant tokens and prune them in early steps. Based on these intuitions, we propose DSAP that employs a **prune-less schedule** in early denoising steps by leaving some of the layers unpruned.

The prune-less schedule. In SD-XL, down-stages, up-stages, and the mid-stage include attention blocks. Each attention block includes 2-10 attention layers. In our prune-less schedule, we select some attention blocks to not perform token pruning. Since previous works [13, 19] indicate that the mid-stage contributes much less to the generated image quality than the up-stages and down-stages, we do not select the attention block in the mid-stage. Based on the ablation study, we choose to leave the first attention block in each down-stage and the last attention block in each up-stage unpruned. We use this prune-less schedule for the first τ denoising steps. We explore setting τ in different regions shown in Fig. 5 and find $\tau = 15$ is the optimal choice. We exhibit all the related ablative experimental results in Section 4.4. A detailed description of the prune-less schedule is provided in Supp. To further consolidate our intuitions, we also investigate a prune-more schedule in early denoising steps and find it inferior to our current approach (Supp).

4. Experimental Results

In this section, we evaluate AT-EDM and ToMe on SD-XL. We provide both visual and quantitative experimental results to demonstrate the advantages of AT-EDM over ToMe.

4.1. Experimental Setup

Common settings. We implement both our AT-EDM method and ToMe on the official repository of SD-XL. The resolution of generated images is 1024×1024 pixels and the default FLOPs budget for each denoising step is assumed to be 4.1T, which is 38.8% smaller than that of the original

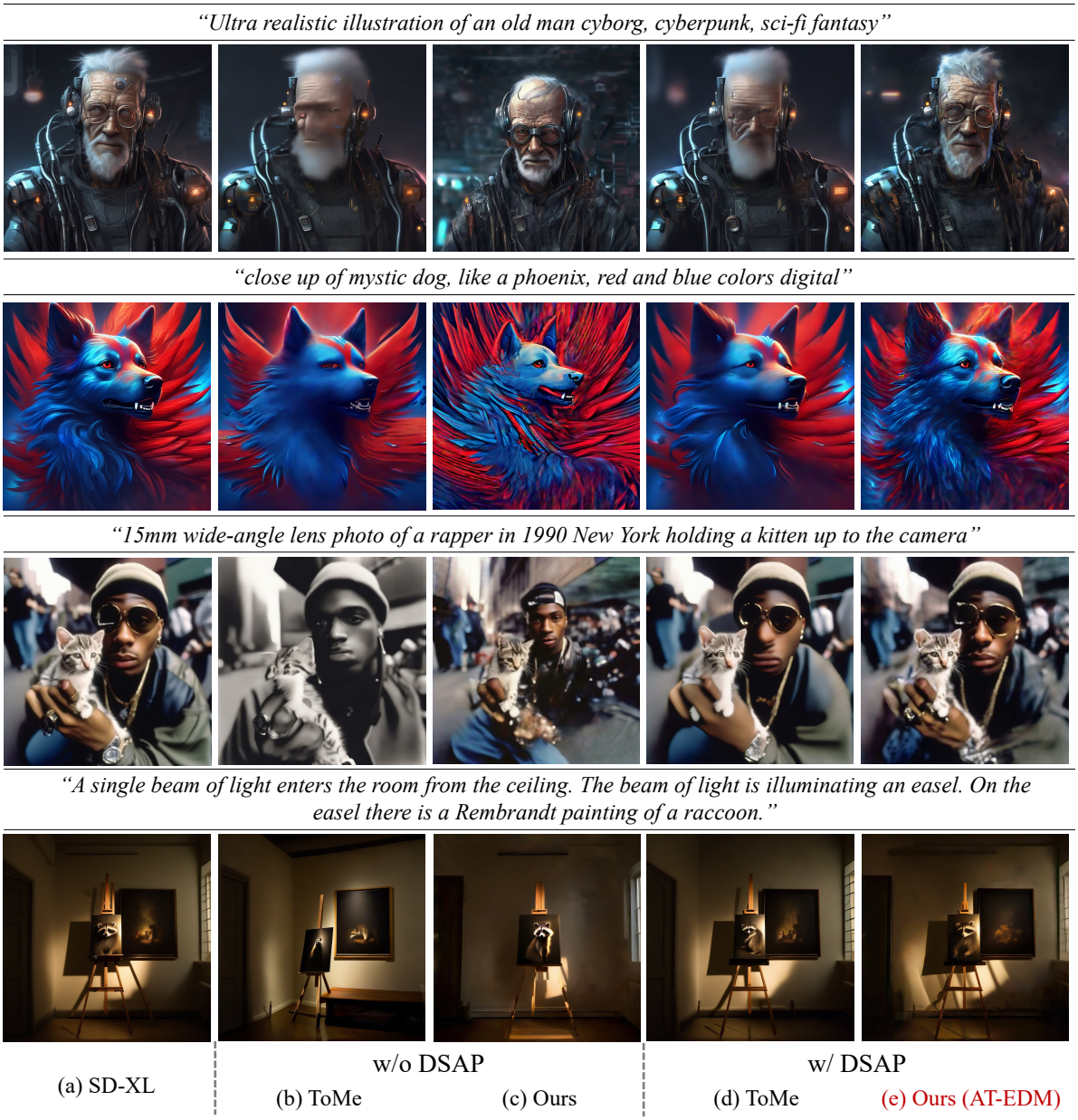

Figure 6. Comparing AT-EDM to the state-of-the-art approach, ToMe [1]. While the full-size SD-XL [26] (**Col. a**) consumes 6.7 TFLOPs, we compare the accelerated models (**Col. b-e**) at the same budget of 4.1 TFLOPs. Compared to ToMe, AT-EDM provides clearer generated objects with sharper details and finer textures, and a better text-image alignment where it better retains the semantics in the prompt (see the fourth row). Moreover, we find that DSAP provides better structural layout of the generated images, which is effective for both ToMe and our approach. AT-EDM combines the novel token pruning algorithm and the DSAP schedule (**Col. e**), outperforming the state of the art.

model (6.7T) unless otherwise noted. The default CFG-scale for image generation is 7.0 unless otherwise noted. We set the total number of sampling steps to 50 and use the default sampler of SD-XL, i.e., EulerEDMSampler.

AT-EDM. For a concise design, we only insert a pruning layer after the first attention layer of each attention block and set the pruning ratio for that layer to ρ. To meet the FLOPs budget of 4.1T, we set $\rho = 63\%$. In the prune-less schedule, we leave the first (last) attention block in each down-stage (up-stage) unpruned. We use this prune-less schedule for the first $\tau = 15$ denoising steps.

ToMe. The SD-XL architecture has changed significantly compared to previous versions of SDMs (see Supp). Thus, the default setting of ToMe does not lead to enough FLOPs savings. To meet the FLOPs budget, it is necessary to use a more aggressive merging setting. Therefore, we expand the application range of token merging (1) from attention layers at the highest feature level to all attention layers, and

(2) from self-attention to self-attention, cross-attention, and the feedforward network. We set the merging ratio $r = 50\%$ to meet the FLOPs budget of 4.1T.

Evaluations. We first compare the generated images with manually designed challenging prompts in Section 4.2. Then, we report FID and CLIP scores of zero-shot image generation on the MS-COCO 2017 validation dataset [20] in Section 4.3. Tested models generate 5k images based on the captions. We provide ablative experimental results and analyze them in Section 4.4 to justify our design choices. We provide more implementation details in Supp.

4.2. Visual Examples for Qualitative Analysis

We use manually designed challenging prompts to evaluate ToMe and our proposed AT-EDM framework. The generated images are compared in Fig. 6. We compare more generated images in Supp. Visual examples indicate that with the same FLOPs budget, AT-EDM demonstrates better **main object preservation** and **text-image alignment** than ToMe. For instance, in the first example, AT-EDM preserves the main object, the face of the old man, much better than ToMe does. AT-EDM's strong ability to preserve the main object is also exhibited in the second example. ToMe loses high-frequency features of the main object, such as texture and hair, while AT-EDM retains them well, even without DSAP. The third example again illustrates the advantage of AT-EDM over ToMe in preserving the rapper's face. The fourth example uses a relatively complex prompt that describes relationships between multiple objects. ToMe misunderstands *"a Rembrandt painting of a raccoon"* as being a random painting on the easel and a painting of a raccoon on the wall. On the contrary, the image generated by AT-EDM understands and preserves these relationships very well, even without DSAP. As a part of our AT-EDM framework, DSAP is not only effective in AT-EDM but also beneficial to ToMe in improving image quality and text-image alignment. When we deploy DSAP in ToMe, we select corresponding attention blocks to not perform token merging, while keeping the FLOPs cost fixed.

4.3. Quantitative Evaluations

FID-CLIP Curves. We explore the trade-off between the CLIP and FID scores through various Classifer-Free Guidance (CFG) scales. We show the results in Fig. 7. AT-EDM† does not deploy pruning at the second feature level (see Supp). It indicates that for most CFG scales, AT-EDM not only lowers the FID score but also results in higher CLIP scores than ToMe, implying that images generated by AT-EDM not only have better quality but also better text-image alignment. Specifically, when the CFG scale equals 7, AT-EDM (ToMe) results in 28.0 (35.3) FID score. Compared with the sweet spot of the full-size model (27.3), **AT-EDM reduces the FID gap from 8.0 to 0.7.**

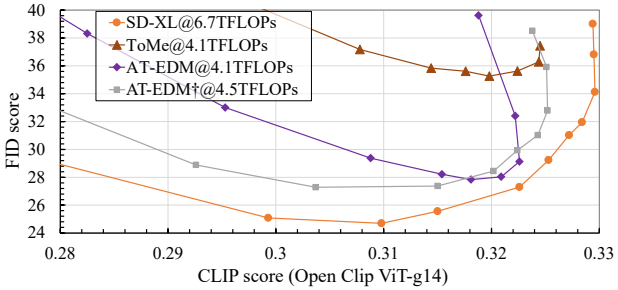

Figure 7. FID-CLIP score curves. The used CFG scales are [1.0, 1.5, 2.0, 2.5, 3.0, 4.0, 5.0, 6.0, 7.0, 9.0, 12.0, 15.0]. This figure is zoomed in to the bottom-right corner to compare the best trade-off points. See complete curves in Supp.

Table 1. Deploying ToMe and AT-EDM in SD-XL under different FLOPs budgets. We generate all images with the CFG-scale of 7.0, except for SD-XL†, for which we use the CFG-scale of 4.0.

Model	FID	CLIP	TFLOPs
SD-XL	31.94	0.3284	6.7
SD-XL†	27.30	0.3226	6.7
ToMe-a	58.76	0.2954	2.9
AT-EDM-a	52.00	0.2784	2.9
ToMe-b	40.94	0.3154	3.6
AT-EDM-b	29.80	0.3095	3.6
ToMe-c	35.27	0.3198	4.1
AT-EDM-c	28.04	0.3209	4.1
ToMe-d	32.46	0.3235	4.6
AT-EDM-d	27.23	0.3245	4.5

Various FLOPs Budgets. We deploy ToMe and AT-EDM on SD-XL under various FLOPs budgets and show the results in Table 1. It indicates that AT-EDM achieves better image quality than ToMe (lower FID scores) under all FLOPs budgets. When the FLOPs saving is 30-40%, AT-EDM achieves not only better image quality (lower FID scores) but also better text-image alignment (higher CLIP scores) than ToMe. Compared to the sweet spot of the full model (CFG-scale equals 4), AT-EDM achieves **not only a lower FID score but also a higher CLIP score while reducing FLOPs by 32.8%**. We provide more visual examples under various FLOPs budgets in Supp.

Latency Analysis. SD-XL uses the Fused Operation (FO) library, xformers [16], to boost its generation. The Current Implementation (CI) of xformers does not provide attention maps as intermediate results; hence, we need to additionally obtain the attention maps. We discuss the sampling latency for three cases: (I) without FO, (II) with FO under CI, and (III) with FO under the Desired Implementation (DI), which provides attention maps as intermediate results. Table 2 shows that with FO, the cost of deploying pruning at the second feature level exceeds the latency reduction it leads to. Hence, AT-EDM† is faster than AT-EDM. We show the extra latency incurred by different pruning steps in Supp. With a negligible quality loss, **AT-EDM achieves 52.7%, 15.4%, 17.6% speed-up in terms of latency w/o FO, w/**

Table 2. Comparison of sampling latency in different cases.

Model Ave. FLOPs/step	SD-XL 6.7 T	ToMe 4.1 T	AT-EDM 4.1 T	AT-EDM† 4.5 T
w/o FO	31.0s	21.0s	**20.3s**	22.1s
w/ FO under CI	18.0s	17.7s	18.3s	**15.6s**
w/ FO under DI	18.0s	17.7s	16.3s	**15.3s**

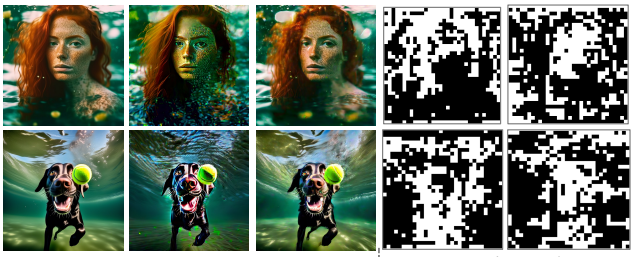

Figure 8. Comparison between implementations of G-WPR: CA-based WPR and SA-based WPR. CA-based WPR may remove too many background tokens, making the background not recoverable, while SA-based WPR preserves the image quality better.

FO under CI, w/ FO under DI, respectively, which outperforms the state-of-the-art work by a clear margin. We provide the memory footprint of AT-EDM in Supp.

4.4. Ablation Study

Self-Attention (SA) vs. Cross-Attention (CA). G-WPR can potentially use attention maps from self-attention (SA-based WPR) and cross-attention (CA-based WPR). We provide a detailed comparison between the two implementations. We visualize their pruning masks and provide generated image examples for a visual comparison in Fig. 8. This figure indicates that SA-based WPR outperforms CA-based WPR. The reason is that CA-based WPR prunes too many background tokens, making it hard to recover the background via similarity-based copy.

Similarity-based Copy. We provide comparisons between different methods to recover the pruned tokens in Fig. 9, which demonstrate the advantages of our similarity-based copy method. Images generated by bicubic interpolation are quite similar to those generated by padding zeros because interpolation usually assigns near-zero values to pruned tokens that are surrounded by other pruned tokens and can hardly recover them. Direct copy means directly copying corresponding token values before the first pruning layer to fill the pruned tokens, where the following attention layers do not process the copied values. Thus, the copied values cannot recover the information in pruned tokens. On the contrary, similarity-based copy uses attention maps and tokens that are retained to recover the pruned tokens, providing significantly higher image quality.

Denoising-Steps-Aware Pruning. We provide ablation experiments on the prune-less schedule design in Supp. Here,

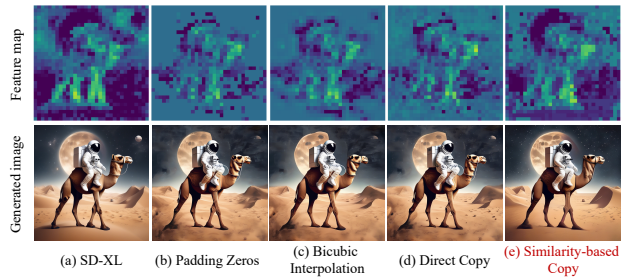

Figure 9. Different methods to recover the pruned tokens. Zero padding, bicubic interpolation, and direct copy can hardly recover pruned tokens and result in noticeable image degradation (incomplete moon). On the contrary, similarity-based copy provides better image quality and keeps the complete moon.

Figure 10. Comparison between different numbers of prune-less steps. Pruning less on the first 15 steps achieves the best quality.

we explore how the number of early prune-less denoising steps affects the generated image quality in Fig. 10. Note that we fix the FLOPs budget and adjust the pruning rate accordingly when we change the number of prune-less steps. This figure shows that the setting of 15 early prune-less steps performs best. Note that the setting of zero prune-less step is identical to the setting without DSAP, and 5, 15, 30, 45 prune-less steps represent setting the boundary in Regions I, II, III, IV of Fig. 5, respectively. The results indicate that placing the boundary between the prune-less and normal schedule in Region II performs best. This meets our expectation because the variance of attention maps becomes high enough to identify unimportant tokens in Region II.

5. Conclusion

We proposed AT-EDM, a novel framework for accelerating DMs at run-time without retraining. In single-denoising-step pruning, AT-EDM exploits attention maps to identify unimportant tokens and prunes them to accelerate the generation process. To solve the compatibility issue, AT-EDM again uses attention maps to reveal similarities between tokens and copies similar tokens to recover the pruned ones. DSAP further improves the generation quality of AT-EDM. Such a pruning schedule is also applicable to other methods like ToMe. Experimental results demonstrate the superiority of AT-EDM with respect to image quality and text-image alignment compared to state-of-the-art methods.

Acknowledgment. This work was supported in part by an Adobe summer internship and in part by NSF under Grant No. CCF-2203399.

References

[1] Daniel Bolya and Judy Hoffman. Token merging for fast stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4598–4602, 2023.

[2] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. *arXiv preprint arXiv:2210.09461*, 2022.

[3] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. MasaCtrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. *arXiv preprint arXiv:2304.08465*, 2023.

[4] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[5] Dave Epstein, Allan Jabri, Ben Poole, Alexei A. Efros, and Aleksander Holynski. Diffusion self-guidance for controllable image generation. *arXiv preprint arXiv:2306.00986*, 2023.

[6] Gongfan Fang, Xinyin Ma, and Xinchao Wang. Structural pruning for diffusion models. *arXiv preprint arXiv:2305.10924*, 2023.

[7] Mohsen Fayyaz, Soroush Abbasi Koohpayegani, Farnoush Rezaei Jafari, Sunando Sengupta, Hamid Reza Vaezi Joze, Eric Sommerlade, Hamed Pirsiavash, and Jürgen Gall. Adaptive token sampling for efficient vision transformers. In *Proceedings of the European Conference on Computer Vision*, pages 396–414. Springer, 2022.

[8] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

[9] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[10] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022.

[11] Jonathan Ho, Chitwan Saharia, William Chan, David J. Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *The Journal of Machine Learning Research*, 23(1):2249–2281, 2022.

[12] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6007–6017, 2023.

[13] Bo-Kyeong Kim, Hyoung-Kyu Song, Thibault Castells, and Shinkook Choi. On architectural compression of text-to-image diffusion models. *arXiv preprint arXiv:2305.15798*, 2023.

[14] Woojeong Kim, Suhyun Kim, Mincheol Park, and Geunseok Jeon. Neuron merging: Compensating for pruned neurons. *Advances in Neural Information Processing Systems*, 33:585–595, 2020.

[15] Woosuk Kwon, Sehoon Kim, Michael W. Mahoney, Joseph Hassoun, Kurt Keutzer, and Amir Gholami. A fast post-training pruning framework for transformers. *Advances in Neural Information Processing Systems*, 35:24101–24116, 2022.

[16] Benjamin Lefaudeux, Francisco Massa, Diana Liskovich, Wenhan Xiong, Vittorio Caggiano, Sean Naren, Min Xu, Jieru Hu, Marta Tintore, Susan Zhang, Patrick Labatut, and Daniel Haziza. xFormers: A modular and hackable transformer modelling library. https://github.com/facebookresearch/xformers, 2022.

[17] Haoying Li, Yifan Yang, Meng Chang, Shiqi Chen, Huajun Feng, Zhihai Xu, Qi Li, and Yueting Chen. SRDiff: Single image super-resolution with diffusion probabilistic models. *Neurocomputing*, 479:47–59, 2022.

[18] Lijiang Li, Huixia Li, Xiawu Zheng, Jie Wu, Xuefeng Xiao, Rui Wang, Min Zheng, Xin Pan, Fei Chao, and Rongrong Ji. AutoDiffusion: Training-free optimization of time steps and architectures for automated diffusion model acceleration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7105–7114, 2023.

[19] Yanyu Li, Huan Wang, Qing Jin, Ju Hu, Pavlo Chemerys, Yun Fu, Yanzhi Wang, Sergey Tulyakov, and Jian Ren. SnapFusion: Text-to-image diffusion model on mobile devices within two seconds. *arXiv preprint arXiv:2306.00980*, 2023.

[20] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, pages 740–755. Springer, 2014.

[21] Enshu Liu, Xuefei Ning, Zinan Lin, Huazhong Yang, and Yu Wang. OMS-DPM: Optimizing the model schedule for diffusion probabilistic models. *arXiv preprint arXiv:2306.08860*, 2023.

[22] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022.

[23] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023.

[24] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik P. Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306, 2023.

[25] Or Patashnik, Daniel Garibi, Idan Azuri, Hadar Averbuch-Elor, and Daniel Cohen-Or. Localizing object-level shape variations with text-to-image diffusion models. *arXiv preprint arXiv:2303.11306*, 2023.

[26] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[27] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. DreamFusion: Text-to-3D using 2D diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

[28] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[29] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[30] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[31] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[32] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.

[33] Suraj Srinivas and R. Venkatesh Babu. Data-free parameter pruning for deep neural networks. *arXiv preprint arXiv:1507.06149*, 2015.

[34] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *Advances in Neural Information Processing Systems*, 34:11287–11302, 2021.

[35] Hongjie Wang, Bhishma Dedhia, and Niraj K. Jha. Zero-TPrune: Zero-shot token pruning through leveraging of the attention graph in pre-trained transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[36] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion GANs. *arXiv preprint arXiv:2112.07804*, 2021.

[37] Shuai Yang, Yukang Chen, Luozhou Wang, Shu Liu, and Yingcong Chen. Denoising diffusion step-aware models. *arXiv preprint arXiv:2310.03337*, 2023.

[38] Xingyi Yang, Daquan Zhou, Jiashi Feng, and Xinchao Wang. Diffusion probabilistic model made slim. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22552–22562, 2023.

[39] Edouard Yvinec, Arnaud Dapogny, Matthieu Cord, and Kevin Bailly. Red: Looking for redundancies for data-free structured compression of deep neural networks. *Advances in Neural Information Processing Systems*, 34:20863–20873, 2021.