

Peer Review

Review of: "Efficient Pruning of Text-to-Image Models: Insights from Pruning Stable Diffusion"

Evgenii Arsentev¹

1. Independent researcher

1. What the work claims

The authors take four post-training pruning methods that were designed for language models (magnitude pruning, Wanda, and each of these with OWL layer-wise sparsity allocation) and apply them to Stable Diffusion 2, pruning the text encoder and the U-Net separately and then together. Quality is measured with FID and CLIP Score on 10,000 images generated from MSCOCO 2017 captions (§3.3, §3.4), with qualitative examples for one prompt per method. Three findings are put forward. First, on the text encoder, plain magnitude pruning beats Wanda at every sparsity (§4.1, Figure 9), and OWL adds little to either. Second, both components show a drop-off point: around 62.5% for the text encoder under magnitude pruning (Figure 1) and a more gradual decline for the U-Net (§4.2, Figure 10). Third, combining the two components at their individual thresholds degrades the model badly (§4.4, Table 6), and backing off in 2.5% steps leads to a recommended configuration of 47.5% text-encoder sparsity and 35% U-Net sparsity, which the paper reports as 38.5% overall sparsity with minimal quality loss.

I like this paper. The question is practical, the design is easy to follow, the component-wise split is the right way to ask it, and the negative result about Wanda is the kind of finding that saves other people time. The code is public (footnote 1). I checked the headline arithmetic using the parameter counts in §3: $(0.475 \times 340M + 0.35 \times 860M) / 1,200M = (161.5M + 301M) / 1,200M = 38.54\%$, which agrees with the 38.5% in the abstract and §4.4.

2. Whether the conclusion holds on the data

The strongest part of the evidence is its scale per configuration. Ten thousand generated images per model is a sensible FID budget (§3.4), and the decision to report both an image-distribution metric and a

text-image alignment metric lets the reader see that the two components fail in different ways (§4.3). The full-model grid in Table 1 is also a careful idea: it fixes overall sparsity and varies only the split, which isolates the allocation question cleanly.

Before going further, I have to state a limit of my own reading. As rendered on Qeios, Tables 2 to 10 appear only as captions; their numbers are not legible in the text I reviewed. The same holds for the curves in Figures 1, 3, 5, 7, 9, 10, and 12. So I could not check any FID or CLIP Score value, and my comments on the results are based on what the prose says about them. I would ask the authors to make sure the tables are present as real tables in the Qeios version.

The direction of CLIP Score. §3.4 says that "a lower CLIP Score indicates a higher correlation in the semantic content of the prompt and image." In the standard CLIPScore of Hessel et al. (reference [40], which the paper cites), higher means better alignment. §4.3 then says that pruning the text encoder "results in a worse CLIP score" and speaks of "the lower CLIP score for text encoder pruning." Under the paper's own definition, a lower score would be better, not worse. Either the definition in §3.4 is a slip, or the implementation is a distance rather than a similarity. Every CLIP Score claim in §4 depends on which it is, so I would fix this first.

Where the U-Net threshold lies. §4.2 states that "50% sparsity is identified as the point where the model's performance significantly deteriorated, and this threshold is later used in full model pruning." The caption of Figure 11 says "Performance greatly suffers beyond 40%", and the same paragraph says performance "steadily decreases as sparsity exceeded 40%." The second approach in §4.4 starts "from the two thresholds" and steps down by 2.5%, ending at 35% for the U-Net. Starting from 50%, that is six steps; from 40%, two. Because the recommended configuration is built on this number, it would help to state one threshold and explain how it was chosen. §4.2 says it was picked "based on a qualitative analysis," which is honest but leaves the reader unable to reproduce the choice.

Single runs and small gaps. The comparison between methods rests on thresholds that differ by one grid step: 62.5% for magnitude pruning against 60% for Wanda (captions of Figures 1 and 3). I could not find seeds, repeated runs, or any interval on FID. With 10,000 images, FID is fairly stable, but generation noise across seeds is not zero, and a 2.5-point difference in where a curve breaks is exactly the kind of gap that a second seed could move. The claim that magnitude pruning wins "at every sparsity level" (§4.1, Figure 9) is likely to survive. The claim that its threshold is later than Wanda's is less certain without repeats. A bootstrap interval on FID, or two extra seeds at the critical sparsities only, would settle it cheaply.

The same caution applies to the anomaly in §4.1, where Wanda's FID at 80% is lower than at 70%, and the text suggests that "Wanda might perform better at higher sparsities." The paper describes images past the magnitude-pruning threshold as "complete noise" (§4.1), and the Figure 4 caption says the Wanda-pruned encoder "breaks down beyond 62.5%". FID between two sets of noise-like images is not a meaningful ranking, and I would present this as a metric artifact rather than as a hint about Wanda.

Why Wanda loses. This is the most interesting result, and it deserves one more control. Wanda scores weights by activation norms collected on calibration inputs (Equation 2). The paper does not say what calibration text was used for the CLIP text encoder, how many samples, or how long they were. If the calibration set was the default C4 setup from the Wanda code, the activations would come from a very different distribution than short image captions. A run with Wanda calibrated on held-out MSCOCO captions would tell us whether the method is badly suited to text-to-image models, as §4.1 concludes, or simply miscalibrated. The same goes for the OWL hyperparameters, which are not given.

What is actually compressed. The abstract and §5 say the configuration is "substantially reducing computational requirements" and opens the way to "resource-constrained devices." Unstructured sparsity, as produced by magnitude pruning and Wanda, does not by itself reduce memory or latency: the zeros still sit in dense tensors unless sparse storage or N:M kernels are used. I found no measurement of model size on disk, peak memory, or generation time in the paper. I would either add such a measurement or soften the wording to "38.5% of weights set to zero".

Arithmetic in Table 1 and §3.2. I recomputed every cell of Table 1 from the parameter counts in §3 (340M text, 860M image). Most cells agree after rounding. Three do not, all in the Text Sparsity column. Row "30%, 75:25": $0.75 \times 360\text{M} / 340\text{M} = 79.4\%$, reported as 80%. Row "30%, 25:75": $0.25 \times 360\text{M} / 340\text{M} = 26.47\%$, reported as 27%. Row "50%, 50:50": $300\text{M} / 340\text{M} = 88.2\%$, reported as 89%. The example in §3.2 says that a 75:25 split at 50% overall sparsity "would result in pruning the text encoder to 177%". From the paper's own counts, it is $0.75 \times 600\text{M} / 340\text{M} = 132.4\%$. The conclusion (infeasible) is unchanged, but the number should be corrected.

The novelty claim. The abstract calls the study "pioneering," and §5 calls it "the first comprehensive study on post-training pruning of text-to-image models." The paper itself cites LD-Pruner [37] and structural pruning for diffusion models [36] in §3.1. I think the narrower claim, the first study of LLM-style one-shot pruning applied separately to the text encoder and the U-Net of Stable Diffusion, is accurate and still interesting. I would use it.

Taken together, the core findings look sound to me. The recommended configuration is reasonable, the Wanda result is real at least under the settings used, and the component analysis is useful. What exceeds the evidence is the language around compute savings, the generality of the claim that "simple magnitude pruning outperforms more advanced techniques" (abstract; only one model, one dataset, one calibration setting), and, in §6, a sentence about SparseGPT that I return to below.

3. Reproducibility

Code is released, which is the most useful thing a paper like this can do, and the dataset is public. Many settings that drive FID, though, are not stated in the text.

- **Model checkpoint.** "Stable Diffusion 2" covers several public checkpoints (2.0 and 2.1, base at 512 px, and the v-prediction model at 768 px). The exact checkpoint name and revision should be given. §2.1 also describes the text encoder as "the CLIP model's text encoder" citing [28]; I would check this against the model card, since Stable Diffusion 2 is usually described as using an OpenCLIP encoder, and say which one was pruned.
- **Sampling.** Sampler, number of denoising steps, guidance scale, output resolution, negative prompt (if any), and seeds are not reported. Each of these moves FID by more than a point.
- **Data split.** §3.3 says 10,000 images were "randomly sampled" from MSCOCO 2017. The 2017 validation split has 5,000 images, so the sample must come from the training split or include several captions per image. Which split, how one caption was chosen per image, and the sampling seed should be stated, ideally with the list of IDs in the repository.
- **Metrics.** The FID implementation (pytorch-fid, clean-fid, torchmetrics), the resizing of real and generated images, and the CLIP backbone used for the CLIP Score are missing. If the CLIP Score backbone is the same family as the pruned text encoder, the bias suggested in §4.3 becomes testable, but only once the backbone is named.
- **Pruning scope.** Which layers were pruned (linear only, or also convolutions in the U-Net; attention projections; embeddings; layer norms) and whether sparsity was per layer or global is not said. Calibration data for Wanda and OWL settings are missing, as noted in section 2.
- **Compute.** Hardware and time per configuration are not given; §4.4 mentions that a full grid search is impractical, and a number would make that concrete.

4. What specifically to fix

1. **§3.4, CLIP Score paragraph, and §4.3.** State whether a higher or lower CLIP Score is better, name the implementation, and make the wording in §4.3 ("worse CLIP score", "lower CLIP score") match. At present, the two sections point in opposite directions.
2. **§4.2 against the Figure 11 caption.** One U-Net threshold, please: 40% or 50%. Then say in §4.4 which value the second approach started from and how many 2.5% steps led to 35%.
3. **Tables 2 to 10.** As rendered on Qeios, none of these tables is legible. Re-upload them as text tables, since they carry all the quantitative evidence.
4. **§3.2, the "177%" example.** With the paper's own 340M/860M split, it is 132.4%. Also correct Table 1, Text Sparsity column, rows "30%, 75:25" (79.4%, not 80%), "30%, 25:75" (26.5%, not 27%), and "50%, 50:50" (88.2%, not 89%), or explain the rounding rule.
5. **§6, first paragraph.** The sentence says that "both Wanda and SparseGPT do not effectively extend to these models." SparseGPT is described in §2.2 but is not among the four methods tested in §3.1. Either add a SparseGPT run on the text encoder or remove it from this sentence.
6. Add a Wanda run calibrated on MSCOCO captions (§4.1), and report the calibration set used for the existing runs.
7. **Abstract and §5:** Replace "substantially reducing computational requirements" with what was measured, or add a short table with checkpoint size under sparse storage, peak memory, and seconds per image for the 38.5% model against the dense one.
8. Report two additional seeds, or a bootstrap interval on FID, at the sparsities around each threshold (Figures 1, 3, 5, 7 near 60% to 62.5%). This would show whether the 2.5-point gap between Wanda and magnitude pruning is stable.
9. **§4.1, Wanda paragraph on 80% sparsity.** I would reframe the lower FID at 80% as an artifact of comparing noise images, and drop the suggestion that Wanda may improve at high sparsity.
10. Name the checkpoint, sampler, steps, guidance scale, resolution, and COCO split in §3 (see section 3 above). A short configuration table is enough.
11. **Abstract, "pioneering", and §5, "the first comprehensive study".** Narrow these to one-shot, LLM-style pruning of Stable Diffusion components, and position the work against [36] and [37] in a sentence.
12. The qualitative figures (2, 4, 6, 8, 11) seem to use one prompt each. A grid of four or five varied prompts at the recommended configuration (Figure 13) would make "minimal quality loss" easier to

judge by eye.

13. Abstract, final sentences: "interoperability" looks like it should be "interpretability", which is the word §5 uses.

5. Recommendation

Minor revisions. This is an honest empirical study with a clear design, adequate sample sizes for FID, released code, and a result about Wanda that is useful to anyone compressing diffusion models. None of the issues I raise undermines the main recommendation of 47.5% and 35% sparsity. The CLIP Score direction, the 40%/50% threshold mismatch, the missing tables on Qeios, and the SparseGPT sentence are quick to correct. The calibration control for Wanda and a latency or memory measurement would make the practical claims much harder to argue with, and I hope the authors add them.

Declarations

Potential competing interests: No potential competing interests to declare.