Supplementary Material of "Will Large-scale Generative Models Corrupt Future Datasets?"

This supplemental material describes experimental settings (Appendix A) and example images from ImageNet and SD-ImageNet (Appendix B).

A. Detailed Experimental Configurations

This section describes the detailed experimental settings and configurations.

A.1. Dataset Creation

We generated images using the StableDiffusion model[4] and its accompanying pre-trained weight (`sd-v1-1.ckpt`) on eight NVIDIA A100 GPUs. Each image of the datasets was sampled by 50 steps of the PLMS sampler with an unconditional guidance scale of 7.5, which is identical to the setting of its web application.[5]

A.2. Image Classification

We trained ResNet-50 and Swin-S models following `torchvision`'s training protocol[6] on eight NVIDIA A100 GPUs. The results of Swin-S were calculated using parameters with exponential moving average.

A.3. Image Captioning

We fine-tuned the captioner and filter modules of the BLIP model following `LAVIS`'s training script[7] on two NVIDIA A100 GPUs.

A.4. Image Generation

We trained and generated images from IDDPM following the official instructions for the ImageNet-64 dataset[8] on eight NVIDIA V-100 GPUs. The model was trained for 1.8×10^6 iterations using the L_{hybrid} objective with a batch size of 512. We generated 50,000 images with 250 sampling steps from EMA models. The computation of metrics is based on https://github.com/NVlabs/stylegan2-ada-pytorch/tree/main/metrics.

A.5. Complex Prompts

We synthetically generated complex prompts for each ImageNet category using the following script.

```
import numpy as np

def generate_prompt(category_names: list[str]) -> str:
    name = np.random.choice(category_names, 1)
    _0 = ['', 'high␣quality', 'low␣quality',
          'monochrome', 'blured', 'atmospheric',
          'rendered', 'zoomed', 'wide-angle',
          'hdr', 'high␣resolution']
    _1 = ['photo', 'picture', 'realistic␣photo',
          'image']
    _2 = ['', 'taken␣with␣iPhone', 'inside',
          'outside', 'without␣background']

    _0 = np.random.choice(_0, None)
    _1 = np.random.choice(_1, None)
    _2 = np.random.choice(_2, None)
    return f"{_0}␣{_1}␣of␣{name}␣{_2}".strip()
```

The words modifying the prompts are selected to mimic human prompts and be applicable to all classes in ImageNet. Because many classes have multiple category names, *e.g.*, "African elephant" and "Loxodonta africana" for the African elephant class, this script can generate a variety of prompts, namely from 200 to 1300 different prompts per class.

A.6. Self-supervised Learning

We pre-trained MAE following the official implementation[9] on eight NVIDIA A-100 GPUs and fine-tuned the last layer of its encoder on 16 NVIDIA V-100 GPUs. Pre-training was for 200 epochs with a 40-epoch warmup with a batch size of 4096, using gradient accumulation once in every two iterations. Fine-tuning was for 90 epochs using the LARS optimizer with a batch size of 16,384.

A.7. Detection of Generated Images

For the experiments in Section 6.1, we first extracted the features of ImageNet-pre-trained ResNet-50 on 12,000 images from ImageNet and SD-ImageNet. Each feature vector has a dimension of 2,048. Then, we trained a linear classifier and a two-layer MLP with a hidden size of 1,024 with a ReLU activation to classify them using 10,000 feature vectors for 5,000 iterations using the Adam optimizer with a batch size of 128. Their performances were evaluated on the other 2,000 test vectors. The linear classifier and the MLP achieve 83% and 86% accuracy, respectively.

B. Additional Results

B.1. Comparison of Real and Generated Images

In Section 4.4, we hypothesized that generated images have fewer modes than real ones, which causes the performance degeneration. Comparing randomly selected images from ImageNet and SD-ImageNet in Figs. B.1 to B.3 visually supports this hypothesis.

[4] https://github.com/CompVis/stable-diffusion

[5] https://huggingface.co/spaces/stabilityai/stable-diffusion/blob/main/app.py

[6] https://github.com/pytorch/vision/tree/v0.13.1/references/classification

[7] https://github.com/salesforce/LAVIS/blob/v0.1.0/run_scripts/blip/train/train_caption_coco.sh

[8] https://github.com/openai/improved-diffusion/tree/main

[9] https://github.com/facebookresearch/mae/tree/main

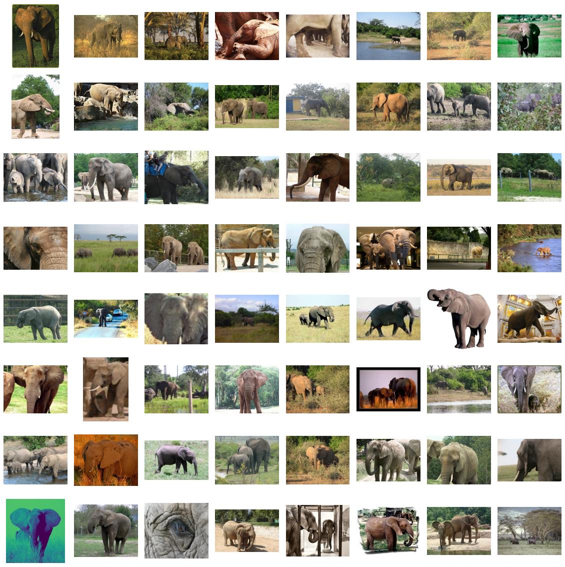

Figure B.1: Real images of African elephants from ImageNet.

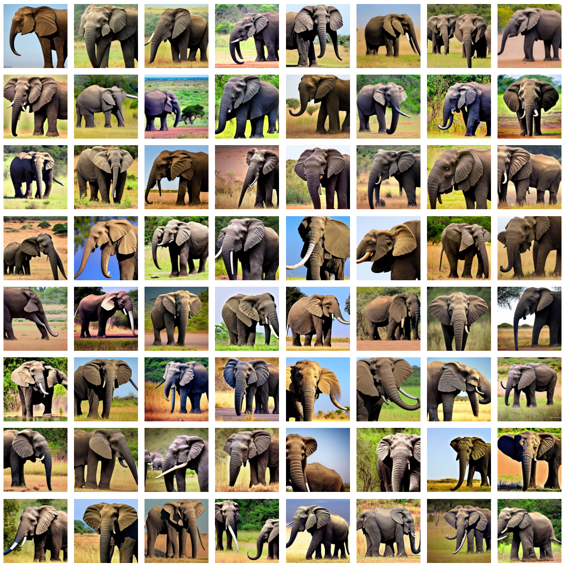

Figure B.2: Generated images of African elephants from the original SD-ImageNet.

B.2. Examples of `titi`

In Section 4.1, we argued that some categories were semantically diverse because the ambiguity of category names. Figure B.4 presents randomly selected images from the `titi` category. Although ImageNet intended this class to mean a New World monkey, the generated images are mostly photos of humans, because "titi" is also a name of people.

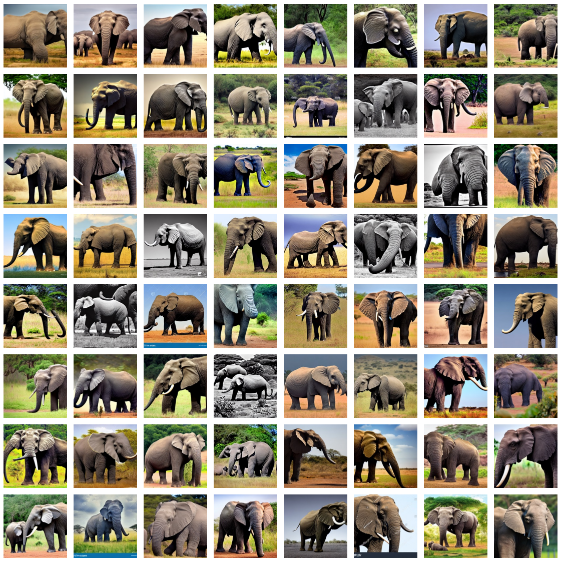

Figure B.3: Generated images of African elephants from the complex SD-ImageNet.

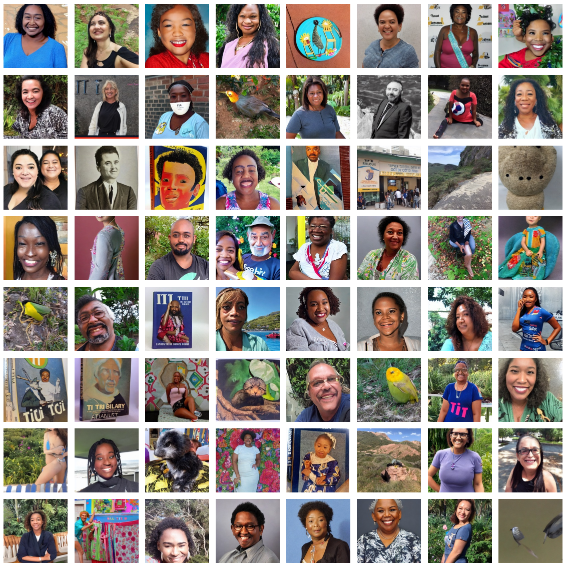

Figure B.4: Generated images of the `titi` category from SD-ImageNet.