Zero-Shot Text-to-Image Generation

Aditya Ramesh [1] Mikhail Pavlov [1] Gabriel Goh [1] Scott Gray [1]
Chelsea Voss [1] Alec Radford [1] Mark Chen [1] Ilya Sutskever [1]

Abstract

Text-to-image generation has traditionally focused on finding better modeling assumptions for training on a fixed dataset. These assumptions might involve complex architectures, auxiliary losses, or side information such as object part labels or segmentation masks supplied during training. We describe a simple approach for this task based on a transformer that autoregressively models the text and image tokens as a single stream of data. With sufficient data and scale, our approach is competitive with previous domain-specific models when evaluated in a zero-shot fashion.

1. Introduction

Modern machine learning approaches to text to image synthesis started with the work of Mansimov et al. (2015), who showed that the DRAW Gregor et al. (2015) generative model, when extended to condition on image captions, could also generate novel visual scenes. Reed et al. (2016b) later demonstrated that using a generative adversarial network (Goodfellow et al., 2014), rather than a recurrent variational auto-encoder, improved image fidelity. Reed et al. (2016b) showed that this system could not only generate objects with recognizable properties, but also could *zero-shot* generalize to held-out categories.

Over the next few years, progress continued using a combination of methods. These include improving the generative model architecture with modifications like multi-scale generators (Zhang et al., 2017; 2018), integrating attention and auxiliary losses (Xu et al., 2018), and leveraging additional sources of conditioning information beyond just text (Reed et al., 2016a; Li et al., 2019; Koh et al., 2021).

Separately, Nguyen et al. (2017) propose an energy-based framework for conditional image generation that obtained a large improvement in sample quality relative to contemporary methods. Their approach can incorporate pretrained discriminative models, and they show that it is capable of performing text-to-image generation when applied to a captioning model pretrained on MS-COCO. More recently, Cho et al. (2020) also propose a method that involves optimizing the input to a pretrained cross-modal masked language model. While significant increases in visual fidelity have occurred as a result of the work since Mansimov et al. (2015), samples can still suffer from severe artifacts such as object distortion, illogical object placement, or unnatural blending of foreground and background elements.

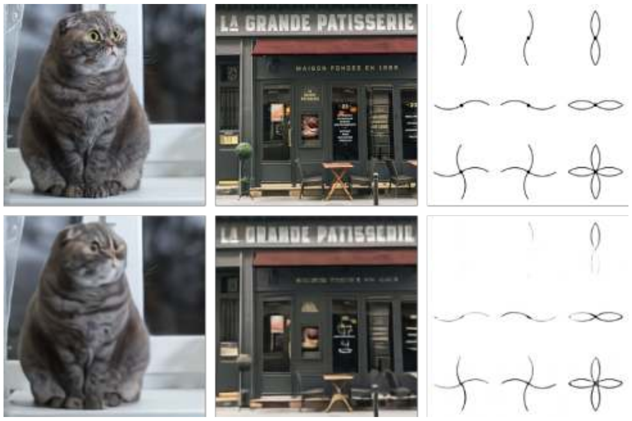


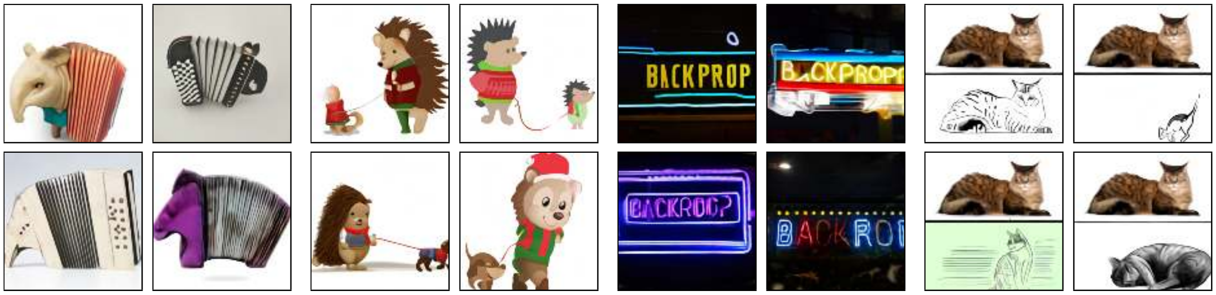
Figure 1. Comparison of original images (top) and reconstructions from the discrete VAE (bottom). The encoder downsamples the spatial resolution by a factor of 8. While details (e.g., the texture of the cat's fur, the writing on the storefront, and the thin lines in the illustration) are sometimes lost or distorted, the main features of the image are still typically recognizable. We use a large vocabulary size of 8192 to mitigate the loss of information.

Recent advances fueled by large-scale generative models suggest a possible route for further improvements. Specifically, when compute, model size, and data are scaled carefully, autoregressive transformers (Vaswani et al., 2017) have achieved impressive results in several domains such as text (Radford et al., 2019), images (Chen et al., 2020), and audio (Dhariwal et al., 2020).

By comparison, text-to-image generation has typically been evaluated on relatively small datasets such as MS-COCO and CUB-200 (Welinder et al., 2010). Could dataset size and

[1] OpenAI, San Francisco, California, United States. Correspondence to: Aditya Ramesh <_@adityaramesh.com>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 139, 2020. Copyright 2020 by the author(s).



(a) a tapir made of accordion. a tapir with the texture of an accordion.

(b) an illustration of a baby hedgehog in a christmas sweater walking a dog

(c) a neon sign that reads "backprop". a neon sign that reads "backprop". backprop neon sign

(d) the exact same cat on the top as a sketch on the bottom

Figure 2. With varying degrees of reliability, our model appears to be able to combine distinct concepts in plausible ways, create anthropomorphized versions of animals, render text, and perform some types of image-to-image translation.

model size be the limiting factor of current approaches? In this work, we demonstrate that training a 12-billion parameter autoregressive transformer on 250 million image-text pairs collected from the internet results in a flexible, high fidelity generative model of images controllable through natural language.

The resulting system achieves high quality image generation on the popular MS-COCO dataset *zero-shot*, without using any of the training labels. It is preferred over prior work trained on the dataset by human evaluators 90% of the time. We also find that it is able to perform complex tasks such as image-to-image translation at a rudimentary level. This previously required custom approaches (Isola et al., 2017), rather emerging as a capability of a single, large generative model.

2. Method

Our goal is to train a transformer (Vaswani et al., 2017) to autoregressively model the text and image tokens as a single stream of data. However, using pixels directly as image tokens would require an inordinate amount of memory for high-resolution images. Likelihood objectives tend to prioritize modeling short-range dependencies between pixels (Salimans et al., 2017), so much of the modeling capacity would be spent capturing high-frequency details instead of the low-frequency structure that makes objects visually recognizable to us.

We address these issues by using a two-stage training procedure, similar to (Oord et al., 2017; Razavi et al., 2019):

- **Stage 1.** We train a discrete variational autoencoder (dVAE)[1] to compress each 256×256 RGB image into a 32×32 grid of image tokens, each element of which can assume 8192 possible values. This reduces the context size of the transformer by a factor of 192 without a large degradation in visual quality (see Figure 1).
- **Stage 2.** We concatenate up to 256 BPE-encoded text tokens with the $32 \times 32 = 1024$ image tokens, and train an autoregressive transformer to model the joint distribution over the text and image tokens.

The overall procedure can be viewed as maximizing the evidence lower bound (ELB) (Kingma & Welling, 2013; Rezende et al., 2014) on the joint likelihood of the model distribution over images x, captions y, and the tokens z for the encoded RGB image. We model this distribution using the factorization $p_{\theta,\psi}(x, y, z) = p_\theta(x \,|\, y, z) p_\psi(y, z)$, which yields the lower bound

$$\ln p_{\theta,\psi}(x, y) \geqslant \underset{z \sim q_\phi(z \,|\, x)}{\mathbb{E}} \Big(\ln p_\theta(x \,|\, y, z) - \beta \, D_{\mathrm{KL}}(q_\phi(y, z \,|\, x), p_\psi(y, z)) \Big), \quad (1)$$

where:

- q_ϕ denotes the distribution over the 32×32 image tokens generated by the dVAE encoder given the RGB image x[2];
- p_θ denotes the distribution over the RGB images generated by the dVAE decoder given the image tokens; and
- p_ψ denotes the joint distribution over the text and image tokens modeled by the transformer.

Note that the bound only holds for $\beta = 1$, while in practice we find it helpful to use larger values (Higgins et al., 2016). The following subsections describe both stages in further detail.[3]

[1] https://github.com/openai/DALL-E

[2] We assume that y is conditionally independent of x given z.

[3] In preliminary experiments on ImageNet (Deng et al., 2009),

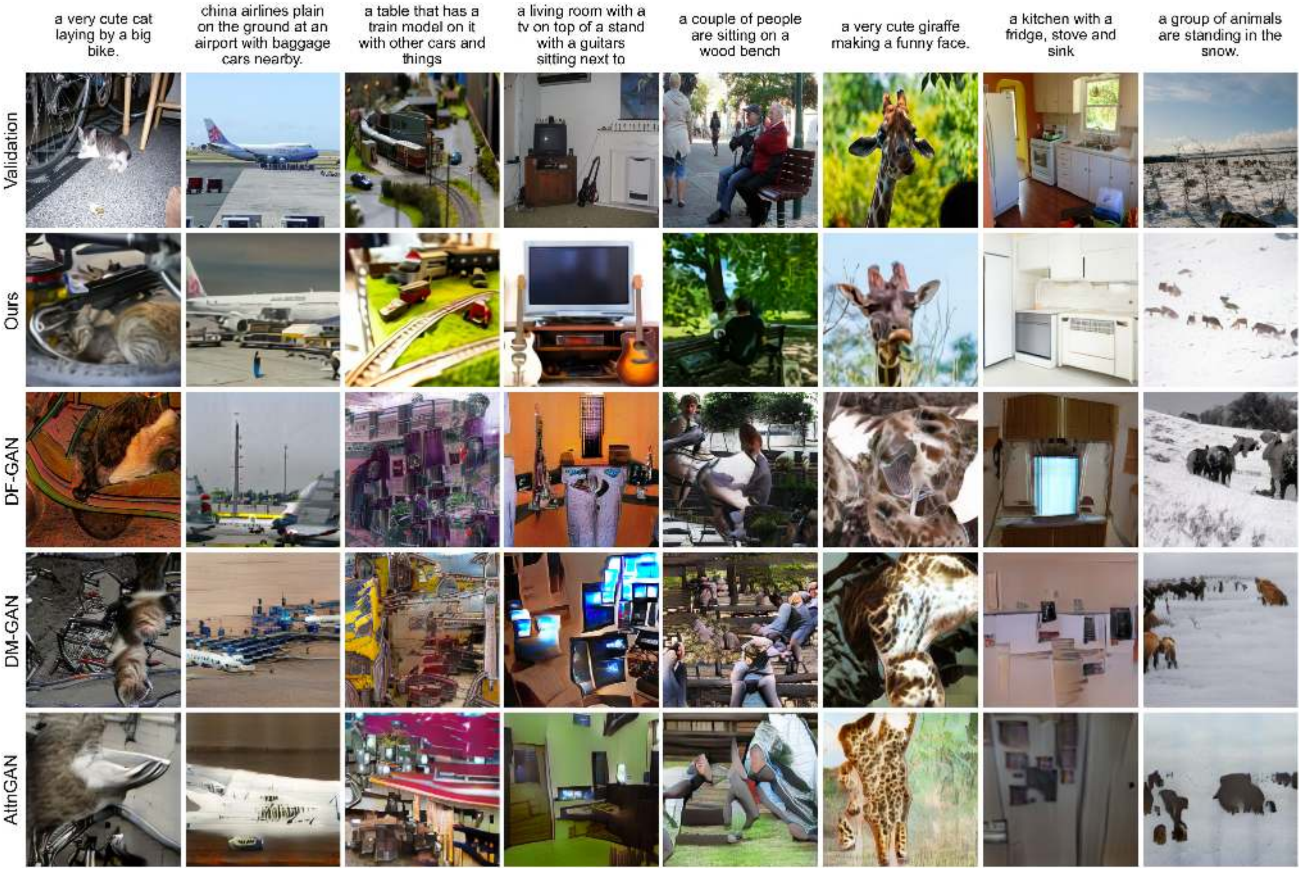

a very cute cat laying by a big bike.
china airlines plain on the ground at an airport with baggage cars nearby.
a table that has a train model on it with other cars and things
a living room with a tv on top of a stand with a guitars sitting next to
a couple of people are sitting on a wood bench
a very cute giraffe making a funny face.
a kitchen with a fridge, stove and sink
a group of animals are standing in the snow.
Validation
Ours
DF-GAN
DM-GAN
AttnGAN

Figure 3. Comparison of samples from our model to those from prior approaches on captions from MS-COCO. Each of our model samples is the best of 512 as ranked by the contrastive model. We do not use any manual cherrypicking with the selection of either the captions or the samples from any of the models.

2.1. Stage One: Learning the Visual Codebook

In the first stage of training, we maximize the ELB with respect to ϕ and θ, which corresponds to training a dVAE on the images alone. We set the initial prior p_ψ to the uniform categorical distribution over the $K = 8192$ codebook vectors, and q_ϕ to be categorical distributions parameterized by the 8192 logits at the same spatial position in the 32×32 grid output by the encoder.

The ELB now becomes difficult to optimize: as q_ψ is a discrete distribution, and we cannot use the reparameterization gradient to maximize it. Oord et al. (2017); Razavi et al. (2019) address this using an online cluster assignment procedure coupled with the straight-through estimator (Bengio et al., 2013). We instead use the gumbel-softmax relaxation (Jang et al., 2016; Maddison et al., 2016), replacing the expectation over q_ϕ with one over q_ϕ^τ, where the relaxation becomes tight as the temperature $\tau \to 0$. The likelihood for p_θ is evaluated using the log-laplace distribution (see Appendix A.3 for a derivation).

The relaxed ELB is maximized using Adam (Kingma & Ba, 2014) with exponentially weighted iterate averaging. Appendix A.2 gives a complete description of the hyperparameters, but we found the following to be especially important for stable training:

- Specific annealing schedules for the relaxation temperature and step size. We found that annealing τ to $1/16$ was sufficient to close the gap between the relaxed validation ELB and the true validation ELB with q_ϕ intsead of q_ϕ^τ.
- The use of 1×1 convolutions at the end of the encoder and the beginning of the decoder. We found that reducing the receptive field size for the convolutions around the relaxation led to it generalizing better to the true ELB.
- Multiplication of the outgoing activations from the encoder and decoder resblocks by a small constant, to ensure stable training at initialization.

We also found that increasing the KL weight to $\beta = 6.6$

we attempted to maximize the ELB with respect to ϕ, θ, and ψ jointly, but were unable to improve on two-stage training.

promotes better codebook usage and ultimately leads to a *smaller* reconstruction error at the end of training.[4]

2.2. Stage Two: Learning the Prior

In the second stage, we fix ϕ and θ, and learn the prior distribution over the text and image tokens by maximizing the ELB with respect to ψ. Here, p_ψ is represented by a 12-billion parameter sparse transformer (Child et al., 2019).

Given a text-image pair, we BPE-encode (Sennrich et al., 2015) the lowercased caption using at most 256 tokens[5] with vocabulary size 16,384, and encode the image using $32 \times 32 = 1024$ tokens with vocabulary size 8192. The image tokens are obtained using argmax sampling from the dVAE encoder logits, without adding any gumbel noise.[6] Finally, the text and image tokens are concatenated and modeled autoregressively as a single stream of data.

The transformer is a decoder-only model in which each image token can attend to all text tokens in any one of its 64 self-attention layers. The full architecture is described in Appendix B.1. There are three different kinds of self-attention masks used in the model. The part of the attention masks corresponding to the text-to-text attention is the standard causal mask, and the part for the image-to-image attention uses either a row, column, or convolutional attention mask.[7]

We limit the length of a text caption to 256 tokens, though it is not totally clear what to do for the "padding" positions in between the last text token and the start-of-image token. One option is to set the logits for these tokens to $-\infty$ in the self-attention operations. Instead, we opt to learn a special padding token separately for each of the 256 text positions. This token is used only when no text token is available. In preliminary experiments on Conceptual Captions (Sharma et al., 2018), we found that this resulted in higher validation loss, but better performance on out-of-distribution captions.

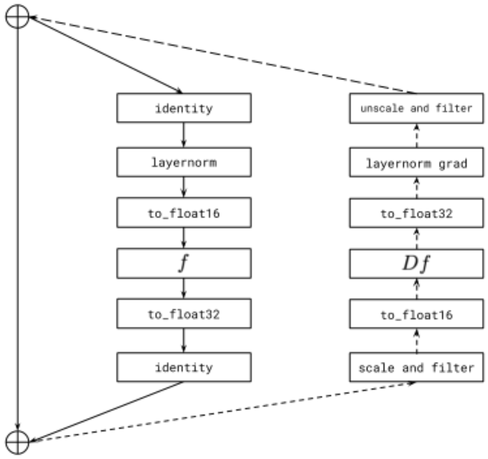


Figure 4. Illustration of per-resblock gradient scaling for a transformer resblock. The solid line indicates the sequence of operations for forward propagation, and the dashed line the sequence of operations for backpropagation. We scale the incoming gradient for each resblock by its gradient scale, and unscale the outgoing gradient before it is added to the sum of the gradients from the successive resblocks. The activations and gradients along the identity path are stored in 32-bit precision. The "filter" operation sets all Inf and NaN values in the activation gradient to zero. Without this, a nonfinite event in the current resblock would cause the gradient scales for all preceding resblocks to unnecessarily drop, thereby resulting in underflow.

We normalize the cross-entropy losses for the text and image tokens by the total number of each kind in a batch of data. Since we are primarily interested in image modeling, we multiply the cross-entropy loss for the text by $1/8$ and the cross-entropy loss for the image by $7/8$. The objective is optimized using Adam with exponentially weighted iterate averaging; Appendix B.2 describes the training procedure in more detail. We reserved about 606,000 images for validation, and found no signs of overfitting at convergence.

2.3. Data Collection

Our preliminary experiments for models up to 1.2 billion parameters were carried out on Conceptual Captions, a dataset of 3.3 million text-image pairs that was developed as an extension to MS-COCO (Lin et al., 2014).

To scale up to 12-billion parameters, we created a dataset of a similar scale to JFT-300M (Sun et al., 2017) by collecting 250 million text-images pairs from the internet. This dataset does not include MS-COCO, but does include Conceptual Captions and a filtered subset of YFCC100M (Thomee et al., 2016). As MS-COCO was created from the latter, our training data includes a fraction of the MS-COCO validation images (but none of the captions). We control for this in the quantitative results presented in Section 3 and find that it has

[4] This is contrary to the usual tradeoff between the two terms. We speculate that for smaller values of β, the noise from the relaxation causes the optimizer to reduce codebook usage toward the beginning of training, resulting in worse ELB at convergence.

[5] During training, we apply 10% BPE dropout (Provilkov et al., 2019), whose use is common in the neural machine translation literature.

[6] Strictly speaking, Equation 1 requires us to sample from the categorical distribution specified by the dVAE encoder logits, rather than taking the argmax. In preliminary experiments on ImageNet, we found that this was a useful regularizer in the overparameterized regime, and allows the transformer to be trained using soft targets for the cross-entropy loss. We decided against this here since the model in consideration is in the underparameterized regime.

[7] We found using a single attention operation for all three interactions – "text attends to text", "image attends to text", and "image attends to image" – to perform better than using separate attention operations that are independently normalized.

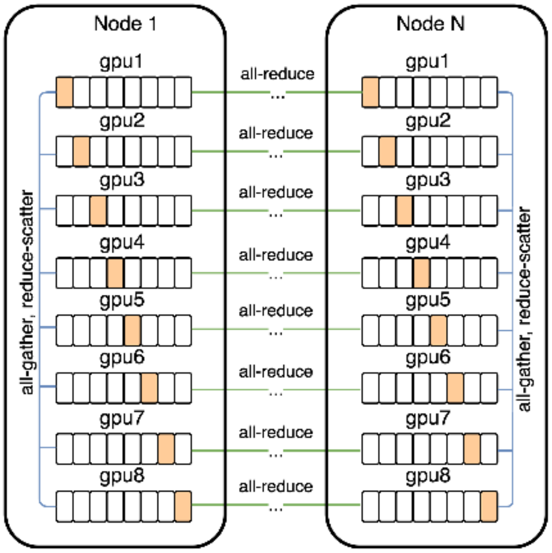


Figure 5. Communication patterns used for distributed training. Each parameter array in the model is sharded among the eight GPUs on each machine. During forward propagation, we prefetch the parameter shards for the next resblock (using all-gather) while computing the activations for the current resblock. To conserve memory, the parameter shards from the other GPUs are immediately discarded. Similarly, during backpropagation, we prefetch the parameter shards for the previous resblock while computing the activations and gradients for the current resblock. After all GPUs have computed the gradient with respect to an all-gathered parameter, the reduce-scatter operation leaves each GPU with only one slice – i.e., the gradient for its parameter shard, averaged over the eight GPUs.

no appreciable bearing on the results. We provide further details about the data collection process in Appendix C.

2.4. Mixed-Precision Training

To save GPU memory and increase throughput, most parameters, Adam moments, and activations are stored in 16-bit precision. We also use activation checkpointing and recompute the activations within the resblocks during the backward pass. Getting the model to train in 16-bit precision past one billion parameters, without diverging, was the most challenging part of this project.

We believe the root cause of this instability to be underflow in the 16-bit gradients. Appendix D presents a set of guidelines we developed to avoid underflow when training large-scale generative models. Here, we describe one of these guidelines: per-resblock gradient scaling.

Similar to prior work (Liu et al., 2020), we found that the norms of the activation gradients from the resblocks decrease monotonically as we move from the earlier resblocks to the later ones.[8] As the model is made deeper and wider, the true exponents of the activation gradients for later resblocks can fall below the minimum exponent of the 16-bit format. Consequently, they get rounded to zero, a phenomenon called *underflow*. We found that eliminating underflow allowed for stable training to convergence.

Effective Parameter Count	Compression Rank	Compression Rate
$2.8 \cdot 10^9$ ($d_{\text{model}} = 1920$)	512	$\approx$83%
$5.6 \cdot 10^9$ ($d_{\text{model}} = 2688$)	640	$\approx$85%
$12.0 \cdot 10^9$ ($d_{\text{model}} = 3968$)	896	$\approx$86%

Table 1. We show the relationship between model size and the minimum compression rank for the gradients (up to a multiple of 128) necessary to avoid a gap in the training loss during the first 10% of training. These results suggest that in our setting, we can achieve a compression rate of about 85%, independent of model size.

Standard loss scaling (Micikevicius et al., 2017) is able to avoid underflow when the range spanned by the smallest and largest activation gradients (in absolute value) fits within the exponent range of the 16-bit format. On NVIDIA V100 GPUs, this exponent range is specified by five bits. While this is sufficient for training vanilla language models of the same size, we found the range to be too small for the text-to-image model.

Our fix, which is shown in Figure 4, involves using a separate "gradient scale" for each resblock in the model. This can be seen as a practical alternative to a more general framework for mixed-precision training called Flexpoint (Köster et al., 2017), with the advantage that specialized GPU kernels are not required. We found that Sun et al. (2020) had independently developed similar procedure for training convolutional networks in 4-bit precision.

2.5. Distributed Optimization

Our 12-billion parameter model consumes about 24 GB of memory when stored in 16-bit precision, which exceeds the memory of a 16 GB NVIDIA V100 GPU. We address this using parameter sharding (Rajbhandari et al., 2019). As shown in Figure 5, parameter sharding allows us to almost completely hide the latency of the intra-machine communication by overlapping it with compute-intensive operations.

On the cluster used to train the model, the bandwidth between machines is much lower than the bandwidth among GPUs on the same machine. This makes the cost of the operation used to average the gradient among the machines (all-reduce) the main bottleneck during training. We were

[8] It is possible that better initialization schemes (Liu et al., 2020) might be able to avoid this, but we did not have success with alternative schemes in our experiments.

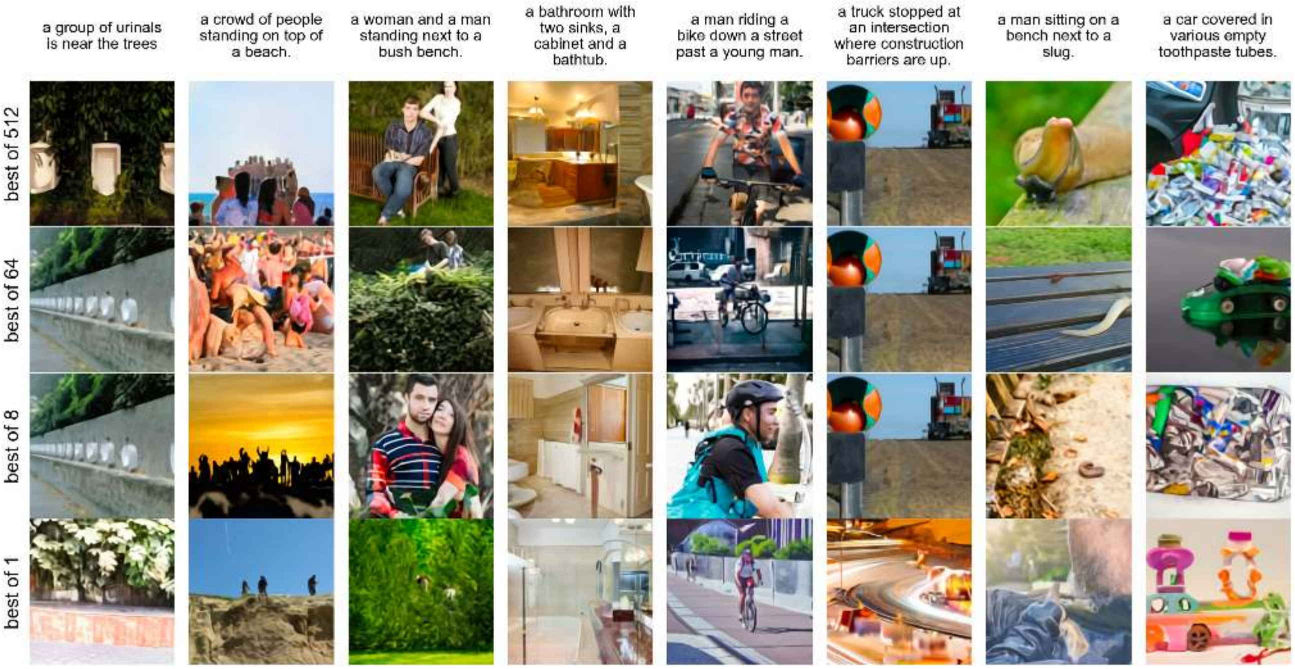

a group of urinals is near the trees
a crowd of people standing on top of a beach.
a woman and a man standing next to a bush bench.
a bathroom with two sinks, a cabinet and a bathtub.
a man riding a bike down a street past a young man.
a truck stopped at an intersection where construction barriers are up.
a man sitting on a bench next to a slug.
a car covered in various empty toothpaste tubes.
best of 512
best of 64
best of 8
best of 1

Figure 6. Effect of increasing the number of images for the contrastive reranking procedure on MS-COCO captions.

able to drastically reduce this cost by compressing the gradients using PowerSGD (Vogels et al., 2019).

In our implementation, each GPU in a machine computes the low-rank factors for its parameter shard gradients independently of its neighboring GPUs.[9] Once the low-rank factors are computed, each machine sets its error buffer to the residual between the uncompressed gradient averaged over its eight GPUs (obtained from reduce-scatter), and the decompressed gradient obtained from the low-rank factors.

PowerSGD replaces the large communication operation for an uncompressed parameter gradient with two, much smaller communication operations for its low-rank factors. For a given compression rank r and transformer activation size d_{model}, the compression rate is given by $1 - 5r/(8d_{\text{model}})$ (see Appendix E.1). Table 1 shows that we can achieve a compression rate of about 85%, independent of model size.

In Appendix E.2, we describe various details that were necessary to get PowerSGD to perform well at scale. These include:

- Saving memory by accumulating the gradient into the error buffers during backpropagation, rather than allocating separate buffers.
- Minimizing instances in which we zero out the error buffers (e.g., due to nonfinite values encountered during mixed-precision backpropagation, or when resuming training from a checkpoint).
- Improving numerical stability by using Householder orthogonalization instead of Gram-Schmidt, together with the addition of a small multiple of the identity matrix to the input.
- Avoiding underflow by using a custom 16-bit floating point format for the error buffers, their low-rank factors, and the all-reduce communication operations involving them.

We also found the warm-start procedure for the Q matrix described in Vogels et al. (2019) to be unnecessary: we were able to get equivalent results by fixing Q to a random gaussian matrix at the start of training, and never updating it.[10]

2.6. Sample Generation

Similar to Razavi et al. (2019), we rerank the samples drawn from the transformer using a pretrained contrastive model (Radford et al., 2021). Given a caption and a candidate image, the contrastive model assigns a score based on

[9] There is still intra-machine communication for other operations; what we mean is that the low-rank factors across the shards, when concatenated, are not regarded as collectively approximating the gradient for the full parameter matrix.

[10] We verified that the error in reconstructing the true gradient is higher when Q is fixed as opposed to being updated using warm-starting, so it is interesting that this does not affect the loss. By contrast, resampling Q at every update causes a large performance hit.

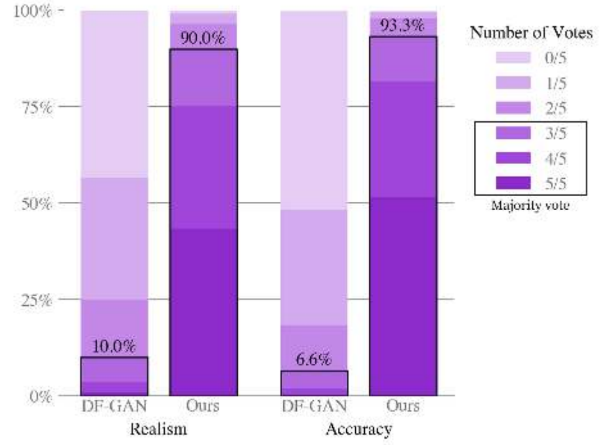


Figure 7. Human evaluation of our model (evaluated zero-shot without temperature reduction) vs prior work (DF-GAN) on captions from MS-COCO. In a best-of-five vote, our model's sample was chosen as the most realistic 90.0% of the time, and was chosen as the image best matching a shared caption 93.3% of the time.

how well the image matches the caption. Figure 6 shows the effect of increasing the number of samples N from which we select the top k images. This process can be seen as a kind of language-guided search (Andreas et al., 2017), and is also similar to the auxiliary text-image matching loss proposed by Xu et al. (2018). Unless otherwise stated, all samples used for both qualitative and quantitative results are obtained without temperature reduction (i.e., using $t = 1$) (except for Figure 2) and use reranking with $N = 512$.

3. Experiments

3.1. Quantitative Results

We evaluate our model zero-shot by comparing it to three prior approaches: AttnGAN (Xu et al., 2018), DM-GAN (Zhu et al., 2019), and DF-GAN (Tao et al., 2020), the last of which reports the best Inception Score (Salimans et al., 2016) and Fréchet Inception Distance (Heusel et al., 2017) on MS-COCO. Figure 3 qualitatively compares samples from our model to those from prior work.

We also conduct a human evaluation similar to the one used in Koh et al. (2021) to compare our approach to DF-GAN, the results of which are shown in Figure 7. Given a caption, the sample from our model receives the majority vote for better matching the caption 93% of the time. It also receives the majority vote for being more realistic 90% of the time.

Figure 9(a) shows that our model also obtains an FID score on MS-COCO within 2 points of the best prior approach, despite having never been trained on the captions. Our training data incorporates a filtered subset of YFCC100M, and we found that it includes about 21% of the images in the MS-COCO validation set from a de-duplication procedure described in the next section. To isolate this effect, we compute the FID statistics for the validation set both with these images (solid lines) and without them (dashed lines), finding no significant change in the results.

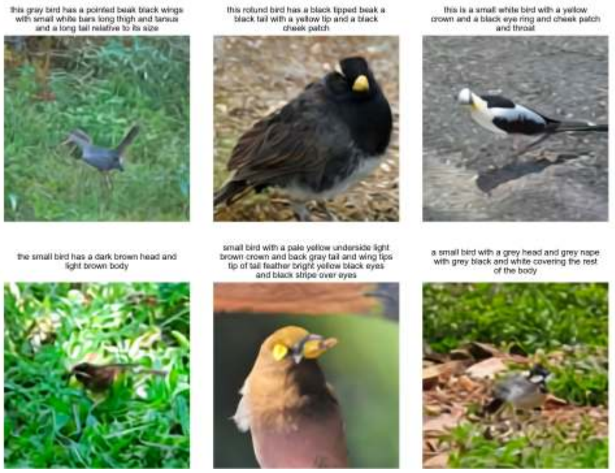
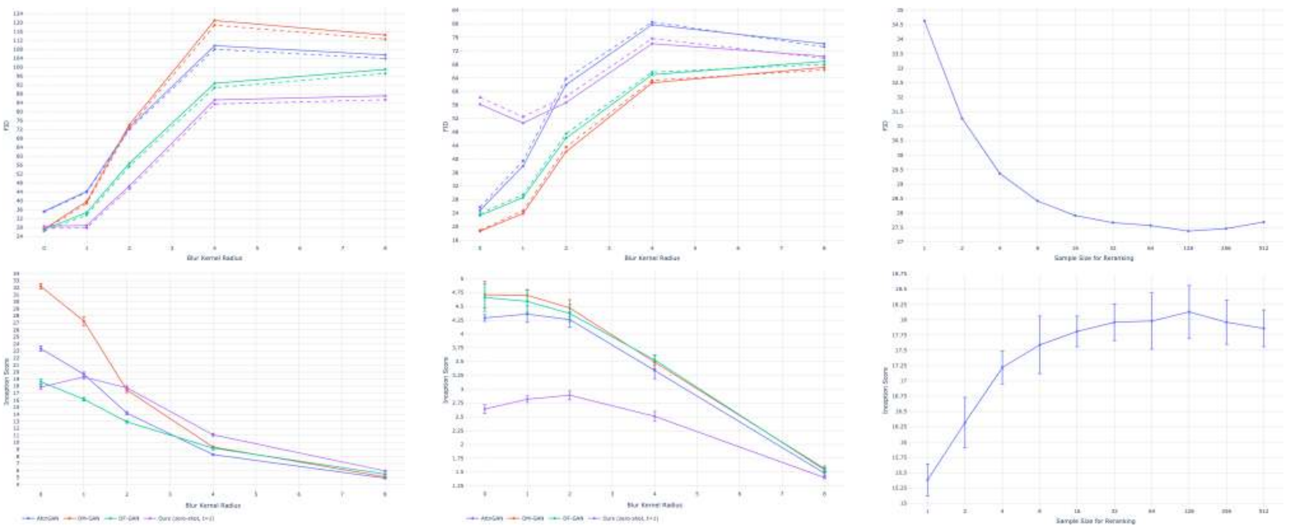

this gray bird has a pointed beak black wings with small white bars long thigh and tarsus and a long tail relative to its size
this rotund bird has a black tipped beak a black tail with a yellow tip and a black cheek patch
this is a small white bird with a yellow crown and a black eye ring and cheek patch and throat
the small bird has a dark brown head and light brown body
small bird with a pale yellow underside light brown crown and back gray tail and wing tips tip of tail feather bright yellow black eyes and black stripe over eyes
a small bird with a grey head and grey nape with grey black and white covering the rest of the body

Figure 8. Zero-shot samples from our model on the CUB dataset.

Training the transformer on the tokens from the dVAE encoder allows us to allocate its modeling capacity to the low-frequency information that makes images visually recognizable to us. However, it also disadvantages the model, since the heavy compression renders it unable to produce high-frequency details. To test the effect of this on the quantitative evaluations, we compute the FID and IS in Figure 9(a) after applying a Gaussian filter with varying radius to both the validation images and samples from the models. Our approach achieves the best FID by a margin of about 6 points with a slight blur of radius 1. The gap between our approach and others tends to widen as the blur radius is increased. We also obtain the highest IS when the blur radius is greater than or equal to two.

Our model fares significantly worse on the CUB dataset, for which there is a nearly 40-point gap in FID between our model and the leading prior approach (Figure 9(b)). We found an 12% overlap rate for this dataset, and again observed no significant difference in the results after removing these images. We speculate that our zero-shot approach is less likely to compare favorably on specialized distributions such as CUB. We believe that fine-tuning is a promising direction for improvement, and leave this investigation to future work. Samples from our model for captions in this dataset are shown in Figure 8.

Finally, Figure 9(c) shows clear improvements in FID and IS for MS-COCO as the sample size used for reranking with the contrastive model is increased. This trend continues up to a sample size of 32, after which we observe diminishing



(a) FID and IS on MS-COCO as a function of blur radius.

(b) FID and IS on CUB as a function of blur radius.

(c) FID and IS on MS-COCO as a function of the sample size used for reranking.

Figure 9. Quantitative results on MS-COCO and CUB. Solid lines represent FID computed against the original validation sets, and dashed lines represent FID computed against validation sets with overlapping images removed (see Section 3.2). For MS-COCO, we evaluate all models on a subset of 30,000 captions sampled from the validation set. For CUB, we evaluate all models on all of the unique captions in the test set. We compute the FID and IS using the DM-GAN code, which is available at `https://github.com/MinfengZhu/DM-GAN`.

returns.

3.2. Data Overlap Analysis

We used the deduplication procedure described in Radford et al. (2021) to determine which images to remove. For each validation image, we find the closest image in the training data using a contrastive model specifically trained for this task. We then sort the images in descending order by closeness to their nearest matches in the training data. After inspecting the results by hand, we determine the images to remove by manually selecting a conservative threshold designed to minimize the false negative rate.

3.3. Qualitative Findings

We found that our model has the ability to generalize in ways that we did not originally anticipate. When given the caption "a tapir made of accordion..." (Figure 2a), the model appears to draw a tapir with an accordion for a body, or an accordion whose keyboard or bass are in the shape of a tapir's trunk or legs. This suggests that it has developed a rudimentary ability to compose unusual concepts at high levels of abstraction.

Our model also appears to be capable of combinatorial generalization, such as when rendering text (Figure 2b) or when probed on sentences like "an illustration of a baby hedgehog in a christmas sweater walking a dog" (Figure 2c). Prompts like the latter require the model to perform variable binding (Smolensky, 1990; Greff et al., 2020) – it is the hedgehog that is in the christmas sweater, not the dog. We note, however, that the model performs inconsistently on the task, sometimes drawing both animals with christmas sweaters, or drawing a hedgehog walking a smaller hedgehog.

To a limited degree of reliability, we also find our model to be capable of zero-shot image-to-image translation controllable by natural language (Figure 2d). When the model is given the caption "the exact same cat on the top as a sketch at the bottom" and the top 15×32 part of the image token grid for a photo of a cat, it is able to draw a sketch of a similar looking cat on the bottom.

This works with several other kinds of transformations, including image operations (e.g., changing the color of the image, converting it to grayscale, or flipping it upside-down) and style transfer (e.g., drawing the cat on a greeting card, a postage stamp, or a cell phone case). Some transformations, such as those that involve only changing the color of the animal, suggest that the model is capable of performing a rudimentary kind of object segmentation. We provide additional examples of zero-shot image-to-image translation in Section G.

4. Conclusion

We investigate a simple approach for text-to-image generation based on an autoregressive transformer, when it is executed at scale. We find that scale can lead to improved generalization, both in terms of zero-shot performance relative to previous domain-specific approaches, and in terms of the range of capabilities that emerge from a single generative model. Our findings suggest that improving generalization as a function of scale may be a useful driver for progress on this task.

Acknowledgements

We would like to thank Matthew Knight for reviewing the code release for this work, and Rewon Child, John Schulman, Heewoo Jun, and Prafulla Dhariwal for helpful early feedback on the paper. We would also like to thank Jong Wook Kim for writing the PyTorch package for the contrastive model described in Radford et al. (2019) that we used to rerank the samples from our model.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pp. 265–283, 2016.

Andreas, J., Klein, D., and Levine, S. Learning with latent language. *arXiv preprint arXiv:1711.00482*, 2017.

Bengio, Y., Léonard, N., and Courville, A. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

Bowman, S. R., Vilnis, L., Vinyals, O., Dai, A. M., Jozefowicz, R., and Bengio, S. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*, 2015.

Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., and Sutskever, I. Generative pretraining from pixels. In *International Conference on Machine Learning*, pp. 1691–1703. PMLR, 2020.

Child, R., Gray, S., Radford, A., and Sutskever, I. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

Cho, J., Lu, J., Schwenk, D., Hajishirzi, H., and Kembhavi, A. X-lxmert: Paint, caption and answer questions with multi-modal transformers. *arXiv preprint arXiv:2009.11278*, 2020.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Dhariwal, P., Jun, H., Payne, C., Kim, J. W., Radford, A., and Sutskever, I. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

Greff, K., van Steenkiste, S., and Schmidhuber, J. On the binding problem in artificial neural networks. *arXiv preprint arXiv:2012.05208*, 2020.

Gregor, K., Danihelka, I., Graves, A., Rezende, D., and Wierstra, D. Draw: A recurrent neural network for image generation. In *International Conference on Machine Learning*, pp. 1462–1471. PMLR, 2015.

He, K., Zhang, X., Ren, S., and Sun, J. Identity mappings in deep residual networks. In *European conference on computer vision*, pp. 630–645. Springer, 2016.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *arXiv preprint arXiv:1706.08500*, 2017.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. 2016.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Koh, J. Y., Baldridge, J., Lee, H., and Yang, Y. Text-to-image generation grounded by fine-grained user attention. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 237–246, 2021.

Köster, U., Webb, T. J., Wang, X., Nassar, M., Bansal, A. K., Constable, W. H., Elibol, O. H., Gray, S., Hall, S., Hornof, L., et al. Flexpoint: An adaptive numerical format for efficient training of deep neural networks. *arXiv preprint arXiv:1711.02213*, 2017.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Li, W., Zhang, P., Zhang, L., Huang, Q., He, X., Lyu, S., and Gao, J. Object-driven text-to-image synthesis via adversarial training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12174–12182, 2019.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Liu, L., Liu, X., Gao, J., Chen, W., and Han, J. Understanding the difficulty of training transformers. *arXiv preprint arXiv:2004.08249*, 2020.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Maddison, C. J., Mnih, A., and Teh, Y. W. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Mansimov, E., Parisotto, E., Ba, J. L., and Salakhutdinov, R. Generating images from captions with attention. *arXiv preprint arXiv:1511.02793*, 2015.

Micikevicius, P., Narang, S., Alben, J., Diamos, G., Elsen, E., Garcia, D., Ginsburg, B., Houston, M., Kuchaiev, O., Venkatesh, G., et al. Mixed precision training. *arXiv preprint arXiv:1710.03740*, 2017.

Nguyen, A., Clune, J., Bengio, Y., Dosovitskiy, A., and Yosinski, J. Plug & play generative networks: Conditional iterative generation of images in latent space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4467–4477, 2017.

Oord, A. v. d., Vinyals, O., and Kavukcuoglu, K. Neural discrete representation learning. *arXiv preprint arXiv:1711.00937*, 2017.

Provilkov, I., Emelianenko, D., and Voita, E. Bpe-dropout: Simple and effective subword regularization. *arXiv preprint arXiv:1910.13267*, 2019.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners. 2019.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. Learning transferable visual models from natural language supervision. 2021.

Rajbhandari, S., Rasley, J., Ruwase, O., and He, Y. Zero: Memory optimization towards training a trillion parameter models. *arXiv preprint arXiv:1910.02054*, 2019.

Razavi, A., Oord, A. v. d., and Vinyals, O. Generating diverse high-fidelity images with vq-vae-2. *arXiv preprint arXiv:1906.00446*, 2019.

Reed, S., Akata, Z., Mohan, S., Tenka, S., Schiele, B., and Lee, H. Learning what and where to draw. *arXiv preprint arXiv:1610.02454*, 2016a.

Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., and Lee, H. Generative adversarial text to image synthesis. In *International Conference on Machine Learning*, pp. 1060–1069. PMLR, 2016b.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pp. 1278–1286. PMLR, 2014.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. Improved techniques for training gans. *arXiv preprint arXiv:1606.03498*, 2016.

Salimans, T., Karpathy, A., Chen, X., and Kingma, D. P. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. *arXiv preprint arXiv:1701.05517*, 2017.

Sennrich, R., Haddow, B., and Birch, A. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*, 2015.

Sharma, P., Ding, N., Goodman, S., and Soricut, R. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2556–2565, 2018.

Smolensky, P. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial intelligence*, 46(1-2):159–216, 1990.

Sun, C., Shrivastava, A., Singh, S., and Gupta, A. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pp. 843–852, 2017.

Sun, X., Wang, N., Chen, C.-Y., Ni, J., Agrawal, A., Cui, X., Venkataramani, S., El Maghraoui, K., Srinivasan, V. V., and Gopalakrishnan, K. Ultra-low precision 4-bit training of deep neural networks. *Advances in Neural Information Processing Systems*, 33, 2020.

Tao, M., Tang, H., Wu, S., Sebe, N., Wu, F., and Jing, X.-Y. Df-gan: Deep fusion generative adversarial networks for text-to-image synthesis. *arXiv preprint arXiv:2008.05865*, 2020.

Thomee, B., Shamma, D. A., Friedland, G., Elizalde, B., Ni, K., Poland, D., Borth, D., and Li, L.-J. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

Vogels, T., Karimireddy, S. P., and Jaggi, M. Powersgd: Practical low-rank gradient compression for distributed optimization. *arXiv preprint arXiv:1905.13727*, 2019.

Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., and Perona, P. Caltech-ucsd birds 200. 2010.

Xu, T., Zhang, P., Huang, Q., Zhang, H., Gan, Z., Huang, X., and He, X. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1316–1324, 2018.

Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., and Metaxas, D. N. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 5907–5915, 2017.

Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., and Metaxas, D. N. Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1947–1962, 2018.

Zhu, M., Pan, P., Chen, W., and Yang, Y. Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5802–5810, 2019.