Google AI

Learning Fast Samplers for Diffusion Models by Differentiating Through Sample Quality

Daniel Watson, William Chan, Jonathan Ho & Mohammad Norouzi

Google AI

Diffusion Probabilistic Models

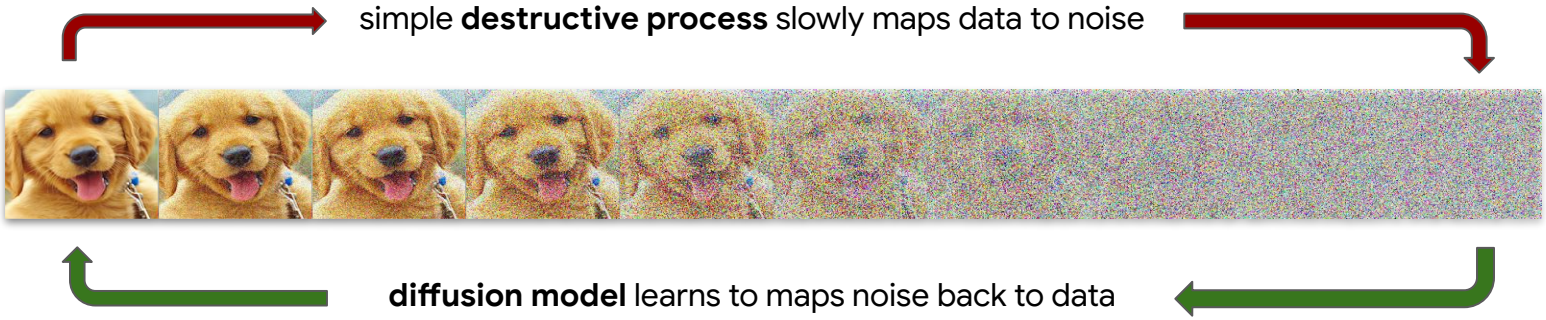


Image credit: Ben Poole

Diffusion Probabilistic Models

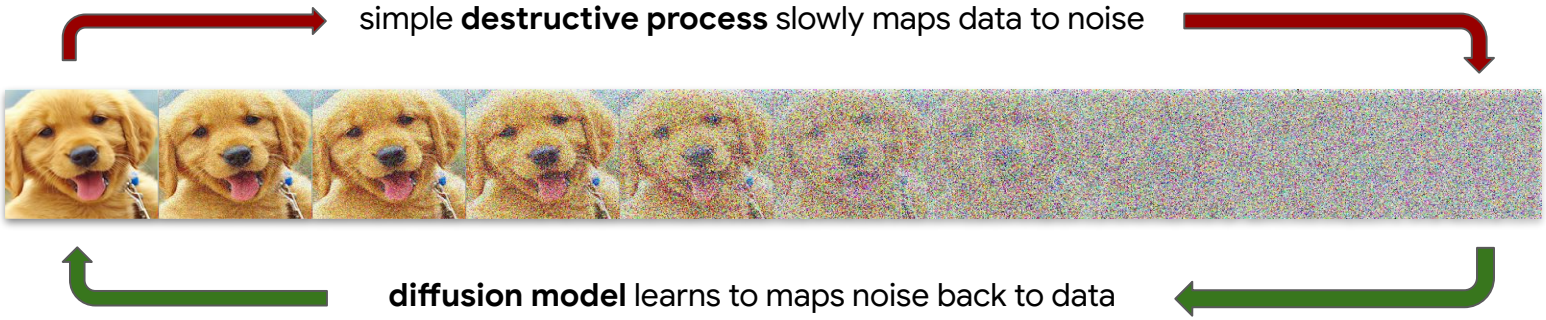


Image credit: Ben Poole

Key problem: for high sample quality, hundreds-to-thousands of denoising steps are needed in practice.

Sampling with few steps with DDIM

Previous SOTA: Denoising Diffusion Implicit Models (DDIM) (Song et al., 2020)

TL;DR. If our Markovian destructive process is defined by

$$q(\mathbf{z}_1, ..., \mathbf{z}_t \mid \mathbf{x}) = q(\mathbf{z}_1 \mid \mathbf{x})q(\mathbf{z}_2 \mid \mathbf{z}_1) ... q(\mathbf{z}_T \mid \mathbf{z}_{T-1}) \text{ where } q(\mathbf{z}_t \mid \mathbf{x}) = N(\mathbf{z}_t; \alpha_t \mathbf{x}, \sigma^2_t I),$$

Sampling with few steps with DDIM

Previous SOTA: Denoising Diffusion Implicit Models (DDIM) (Song et al., 2020)

TL;DR. If our Markovian destructive process is defined by

$q(\mathbf{z}_1, ..., \mathbf{z}_t \mid \mathbf{x}) = q(\mathbf{z}_1 \mid \mathbf{x})q(\mathbf{z}_2 \mid \mathbf{z}_1) ... q(\mathbf{z}_T \mid \mathbf{z}_{T-1})$ where $q(\mathbf{z}_t \mid \mathbf{x}) = N(\mathbf{z}_t; \alpha_t \mathbf{x}, \sigma^2_t I)$,

then this family of non-Markovian processes q_λ parametrized by $(\lambda_1, ..., \lambda_T)$:

$q_\lambda(\mathbf{z}_1, ..., \mathbf{z}_t \mid \mathbf{x}) = q(\mathbf{z}_T \mid \mathbf{x})q_\lambda(\mathbf{z}_{T-1} \mid \mathbf{z}_T, \mathbf{x}) ... q_\lambda(\mathbf{z}_1 \mid \mathbf{z}_2, \mathbf{x})$ where

$q_\lambda(\mathbf{z}_{t-1} \mid \mathbf{z}_t, \mathbf{x}) = N(\mathbf{z}_t; \alpha_{t-1}\mathbf{x} + (\sigma^2_{t-1} - \lambda^2_t)^{1/2} (\mathbf{z}_t - \alpha_t \mathbf{x})/\sigma_t, \lambda^2_t I)$

satisfies $q_\lambda(\mathbf{z}_t \mid \mathbf{x}) = q(\mathbf{z}_t \mid \mathbf{x})$ for all $\lambda_t \in [0, \sigma^2_{t-1}]$.

Sampling with few steps with DDIM

Previous SOTA: Denoising Diffusion Implicit Models (DDIM) (Song et al., 2020)

$q_\lambda(\mathbf{z}_1, ..., \mathbf{z}_t \mid \mathbf{x}) = q(\mathbf{z}_T \mid \mathbf{x}) q_\lambda(\mathbf{z}_{T-1} \mid \mathbf{z}_T, \mathbf{x}) \dots q_\lambda(\mathbf{z}_1 \mid \mathbf{z}_2, \mathbf{x})$ where
$q_\lambda(\mathbf{z}_{t-1} \mid \mathbf{z}_t, \mathbf{x}) = N(\mathbf{z}_t; \alpha_{t-1}\mathbf{x} + (\sigma^2_{t-1} - \lambda^2_t)^{1/2} (\mathbf{z}_t - \alpha_t \mathbf{x})/\sigma_t, \lambda^2_t I)$

satisfies $q_\lambda(\mathbf{z}_t \mid \mathbf{x}) = q(\mathbf{z}_t \mid \mathbf{x})$ for all $\lambda_t \in [0, \sigma^2_{t-1}]$.

- Diffusion training only depends on these marginals, and all DDIM marginals match the usual DDPM's marginals. Hence, all samplers that they define, which are constructed from $q_\lambda(\mathbf{z}_{t-1} \mid \mathbf{z}_t, \mathbf{x})$, are "compatible" with the pre-trained diffusion model.

- The authors empirically find that the choice $\lambda_t = 0$ (for all t) leads to better few-step samplers.

Differentiable Diffusion Sampler Search (DDSS)

Given *any* parametric family of samplers for a *fixed* pre-trained diffusion model, can we just search which choice of such family's degrees of freedom leads to the best sample quality?

Key observation: using the reparametrization trick, we can backprop through the entire sampler, and optimize these degrees of freedom using certain sample quality scores as our objective, e.g., the Kernel Inception Distance (KID) (Bińkowski et al., 2018).

In other words, we can perform this search with gradient-based optimization!

Differentiable Diffusion Sampler Search (DDSS)

- How to backprop without running out of memory?
  Rematerialize (just) the UNet forward pass at each iteration of the sampling chain. O(T * fw_pass_time) extra computation to save O(T * model_state_size) memory.

  If the UNet forward pass is `model_fn = lambda x_t, t: …`

  all you need is to replace `model_fn` with `jax.remat(model_fn)` in your sampling loop.

Experiments

- We experimented with different kernels for KID, and empirically found that DDSS is quite robust to the choice of kernel. Still, we found the simplest (linear) kernel

  $k_\phi(\mathbf{x'}, \mathbf{x}) = \phi(\mathbf{x'}) \cdot \phi(\mathbf{x})$

  yielded the best results, where $\phi(x)$ the previous-to-last activations of a pre-trained image classifier.

- Unbiased estimator of KID: for a parametric family of reverse processes p_ν with parameters ν, N model samples $\mathbf{x'}_i \sim p_\nu$, as well as a data distribution $q(\mathbf{x})$ and N real samples $\mathbf{x}_i \sim q$,

  $L_{KID} = 1 / (N(N-1)) \Sigma_{i \neq j} k_\phi(\mathbf{x'}_i, \mathbf{x}_j) - 2 / N^2 \Sigma_i \Sigma_j k_\phi(\mathbf{x'}_i, \mathbf{x}_j)$

Experiments

- We obtain our best results by optimizing the degrees of freedom of sampler families that generalize DDIM and still yield Gaussian marginals and posteriors.

We define the **Relaxed DDIMs (RDDIM)** parametric family as a joint distribution with parameters μ_t, ν_t, σ_t for each $t \in \{1, ..., T\}$ and the same independence assumptions as DDIMs, i.e.,

$$q_{\mu,\nu,\sigma}(\boldsymbol{x}_0, ..., \boldsymbol{x}_T) = q(\boldsymbol{x}_0)q(\boldsymbol{x}_T|\boldsymbol{x}_0) \prod_{t=1}^{T-1} q_{\mu,\nu,\sigma}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) \tag{10}$$

where the new factors are defined for each t as

$$q_{\mu,\nu,\sigma}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\left(\boldsymbol{x}_{t-1}\middle|\mu_t \boldsymbol{x}_0 + \nu_t \boldsymbol{x}_t, \sigma_t^2 \boldsymbol{I}_d\right) \tag{11}$$

Experiments

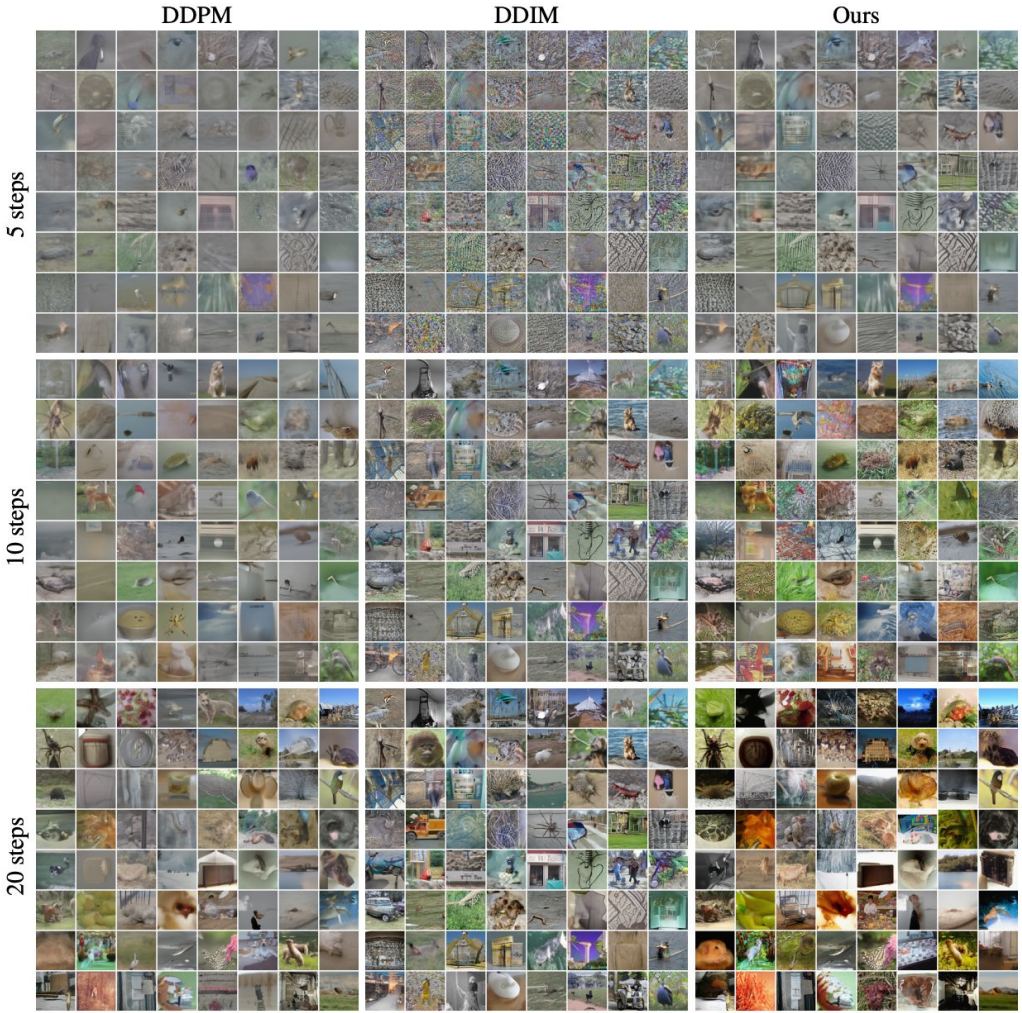
- We obtain our best results by optimizing the degrees of freedom of sampler families that generalize DDIM and still yield Gaussian marginals and posteriors.

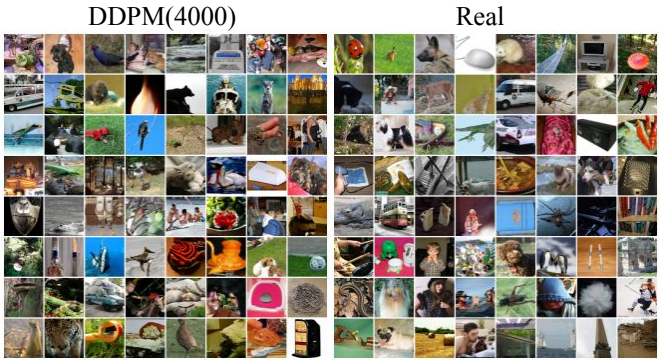
We define **Generalized DDIMs (GDDIMs)** as follows:

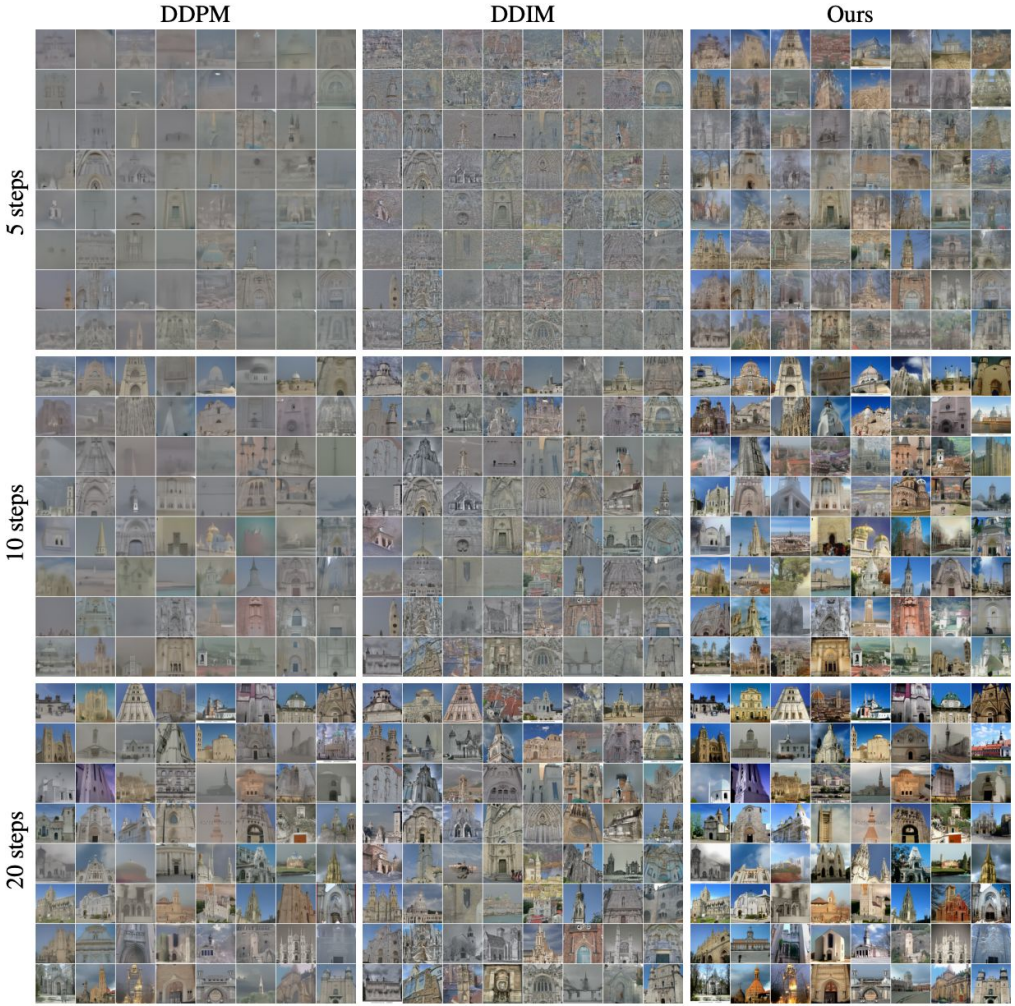
$$q_{\mu,\sigma}(\boldsymbol{x}_0, ..., \boldsymbol{x}_T) = q(\boldsymbol{x}_0)q(\boldsymbol{x}_T|\boldsymbol{x}_0) \prod_{t=1}^{T-1} q_{\mu,\sigma}(\boldsymbol{x}_t|\boldsymbol{x}_{>t}, \boldsymbol{x}_0) \tag{13}$$

where the new factors are defined as

$$q_{\mu,\sigma}(\boldsymbol{x}_t|\boldsymbol{x}_{>t}, \boldsymbol{x}_0) = \mathcal{N}\left(\boldsymbol{x}_t \middle| \sum_{u \in S_t} \mu_{tu}\boldsymbol{x}_u, \sigma_t^2 \boldsymbol{I}_d\right) \tag{14}$$


DDPM
DDIM
Ours
5 steps
10 steps
20 steps


DDPM(4000)
Real



Sampler \ K	5	10	20	1000
CIFAR10 32x32				
DDPM	60.71	31.13	16.08	3.065
DDIM	35.02	12.39	7.442	–
DDSS (ours)	**15.97**	**6.358**	**3.320**	–
ImageNet 64x64				
DDPM	**71.29**	59.36	38.42	16.58
DDIM	254.8	127.6	44.54	–
DDSS (ours)	80.45	**46.98**	**26.61**	–
LSUN Bedroom 128x128				
DDPM	95.38	44.84	16.88	2.457
DDIM	168.7	56.33	9.527	–
DDSS (ours)	**29.20**	**11.02**	**4.854**	–
LSUN Church 128x128				
DDPM	96.67	51.05	16.53	2.718
DDIM	133.1	54.39	14.96	–
DDSS (ours)	**30.14**	**11.60**	**6.651**	–

Summary of contributions

- We propose **Differentiable Diffusion Sampler Search (DDSS)**, an optimization-based procedure to extract a high-fidelity sampler from a parametric family of samplers.

- We can search for a high-fidelity sampler by optimizing KID directly, and show it's possible to differentiate through the whole sampler with the reparametrization trick + gradient rematerialization.

- We propose more general, non-Markovian parametric families of samplers where better solutions lie.

- Our method is compatible with any pre-trained diffusion model, and it does not finetune it nor require re-training it.