


# Discrete Diffusion Modeling by Estimating the Ratios of the Data Distribution

Aaron Lou, Chenlin Meng, Stefano Ermon



# Large Language Models

 ChatGPT  
Certainly! Below is a simple Python function that takes a list of numbers and returns a list of running averages with a window size of 100:

```
python Copy code  
  
def running_averages(input_list):  
    if len(input_list) < 100:  
        raise ValueError("Input list should have at least 100 values.")
```



All are deep generative models on discrete spaces!

# Problems with Discrete Probabilistic Modeling

1.  $p_\theta(x) \geq 0$
2.  $\sum_{x \in \mathcal{X}} p_\theta(x) = 1$

$$p_\theta = e^{f_\theta} / \boxed{Z} \quad \text{Energy Based Model}$$

“Normalizing Constant”

“Partition Function”

“Permanent”

**THE COMPLEXITY OF COMPUTING THE PERMANENT**

**L.G. VALIANT**

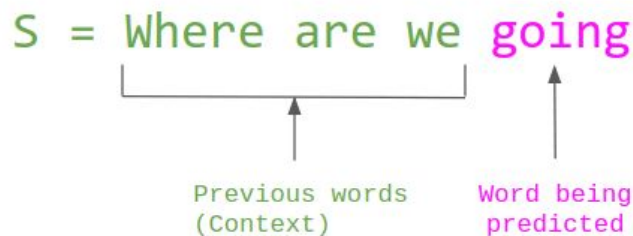
*Computer Science Department, University of Edinburgh, Edinburgh EH9 3JZ, Scotland*

Communicated by M.S. Paterson

Received October 1977

**Unbiased estimators of partition functions are  
basically lower bounds**

# Best Approach So Far: Autoregressive Modeling



$$p_{\theta}(\mathbf{x}) = p_{\theta}(x^1 x^2 \dots x^d) \quad \text{unscalable}$$
$$= p_{\theta}(x^1) p_{\theta}(x^2 | x^1) \dots p_{\theta}(x^d | x^1 x^2 \dots x^{d-1})$$

scalable      scalable      scalable

# Autoregressive Modeling - Upsides

- ✓ Can theoretically represent any probability value
- ✓ Seems to be a reasonable inductive bias for language
- ✗ Hard to control (e.g. editing capabilities, infilling)
- ✗ Doesn't incorporate global information

# Rethinking the Problem

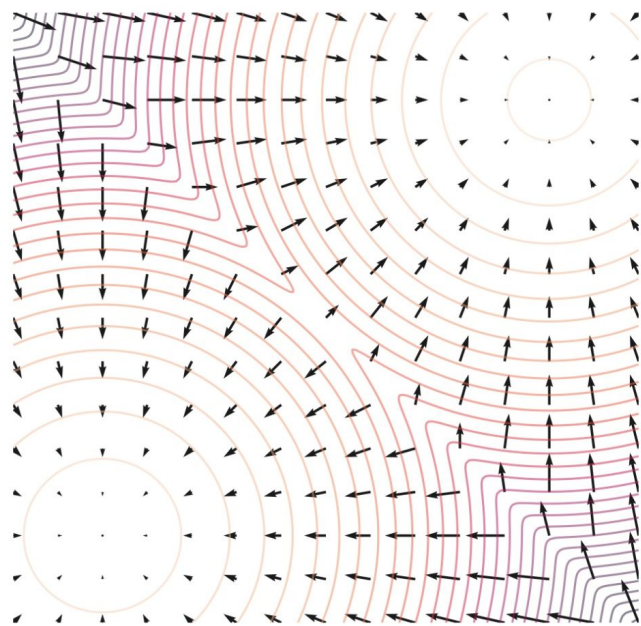
Problem: modeling  $p_{\theta}(\mathbf{x})$  is extremely hard!

Potential Solution: modeling  $\frac{p_{\theta}(\mathbf{y})}{p_{\theta}(\mathbf{x})}$  is easy!

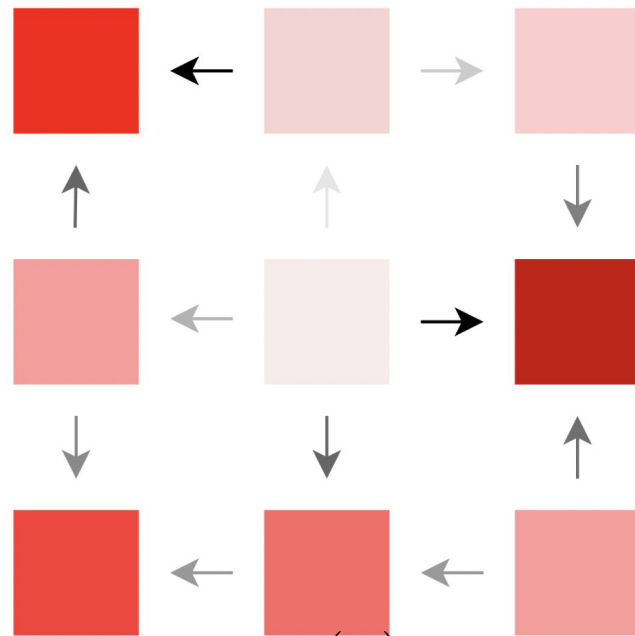
$$\frac{p_{\theta}(\mathbf{y})}{p_{\theta}(\mathbf{x})} = \frac{e^{f_{\theta}(\mathbf{y})} / \cancel{\Sigma}}{e^{f_{\theta}(\mathbf{x})} / \cancel{\Sigma}}$$

Concrete Score

# Why is it called the Concrete Score?



$$\nabla_x \log p_t$$



$$\frac{p_\theta(\mathbf{y})}{p_\theta(\mathbf{x})}$$

# Learning Concrete Scores with Score Entropy

Goal: learn a neural network  $s_\theta(x)$  s.t.  $s_\theta(x)_y \approx \frac{p(y)}{p(x)}$

$$\min_{\theta} \mathbb{E}_{x \sim p} \sum_{y \neq x} s_\theta(x)_y - \frac{p(y)}{p(x)} \log s_\theta(x)_y$$



# Learning Concrete Scores with Score Entropy

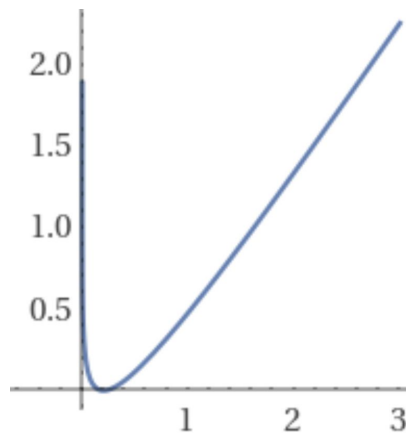
$$\min_{\theta} \mathbb{E}_{x \sim p} \sum_{y \neq x} s_{\theta}(x)_y - \frac{p(y)}{p(x)} \log s_{\theta}(x)_y$$

$$\min \left( s - \frac{p(y)}{p(x)} \log s \right)$$

$$\implies \left( s - \frac{p(y)}{p(x)} \log s \right)' = 0$$

$$\implies 1 - \frac{p(y)}{p(x)} \frac{1}{s} = 0$$

$$\implies s = \frac{p(y)}{p(x)}$$



# Denosing Score Entropy

$$\text{Assume } p(x) = \sum_{x_0} p(x|x_0)p_0(x_0)$$

$$\mathbb{E}_{x \sim p} \sum_{y \neq x} \frac{p(y)}{p(x)} \log s_{\theta}(x)_y = \sum_x \sum_{y \neq x} \log s_{\theta}(x)_y p(y)$$

Hard part of  
SE Loss

$$= \sum_x \sum_{y \neq x} \log s_{\theta}(x)_y \sum_{x_0} p(y|x_0)p_0(x_0)$$

$$= \sum_{x_0} \sum_x \sum_{y \neq x} \log s_{\theta}(x)_y \frac{p(y|x_0)}{p(x|x_0)} p(x|x_0)p_0(x_0)$$

$$= \mathbb{E}_{x_0 \sim p_0, x \sim p(\cdot|x_0)} \sum_{y \neq x} \frac{p(y|x_0)}{p(x|x_0)} \log s_{\theta}(x)_y$$

# Denoising Score Entropy - (cont.)

$$\mathbb{E}_{\substack{x_0 \sim p_0, \\ x \sim p(\cdot | x_0)}} \sum_{y \neq x} s_{\theta}(x)_y - \frac{p(y|x_0)}{p(x|x_0)} \log s_{\theta}(x)_y$$

Sampled      Sampled       $y \neq x$       Compute  $s_{\theta}(x)$  once      Computable

# Continuous Time Markov Chains

Evolutions of the data distribution  $p_0$  :

$$dp_t = Q_t p_t$$

Columns of  $Q_t$  control how often one changes state.

$$p(x_{t+\Delta t} = j | x_t = i) = \delta_{i,j} + \boxed{Q_t(j, i)} \Delta t + O(\Delta t^2)$$

Jump  
transition rate  
from i to j.

# Reversing a Markov Chain

Assume we perturb from  $p_0 \approx p_{\text{data}}$  to  $p_T \approx p_{\text{base}}$

Can we go from  $p_T \approx p_{\text{base}}$  to  $p_0 \approx p_{\text{data}}$ ?

$$dp_{T-t} = \overline{Q}_{T-t} p_{T-t}$$

$$\overline{Q}_t(j, i) = \frac{p_t(j)}{p_t(i)} Q_t(i, j)$$

Learned through  
score entropy

# Putting it all together

1. Get samples from desired data distribution
2. Define a forward diffusion process
3. Learn ratios using Denoising Score Entropy
4. Reverse diffusion process (possibly with some discretization).

# Reversing a Markov Chain - Examples

study ants bear burrito Stanford song

MASK MASK MASK MASK MASK MASK

# Putting it all together

Wyman worked as a computer science coach before going to work with the U.S. Secret Service in upstate New York in 2010. Without a license, the Secret Service will have to oversee both the analysts on the software.

“I see this as going to be a matter of choice, but it has been a long road,” said Mark McSmith, who specializes in the management of data privacy in the National Security Administration. That includes similar uncertainty about what software must be followed and confidentiality rules under the Espionage Act.

Though the software only takes about four years, he said, for the government to get a license for it, it could take after a federal employee spent a while.

“I think I had to read a lot that nobody was telling the Justice Department about it,” he said, adding that “I would guess that it was acquired more recently.” But the company lobbied the feds so it could instead oversee its project using a government arm, because of the Bureau of Law.

Do denied the inquiry, and said it made numerous attempts to be in compliance.

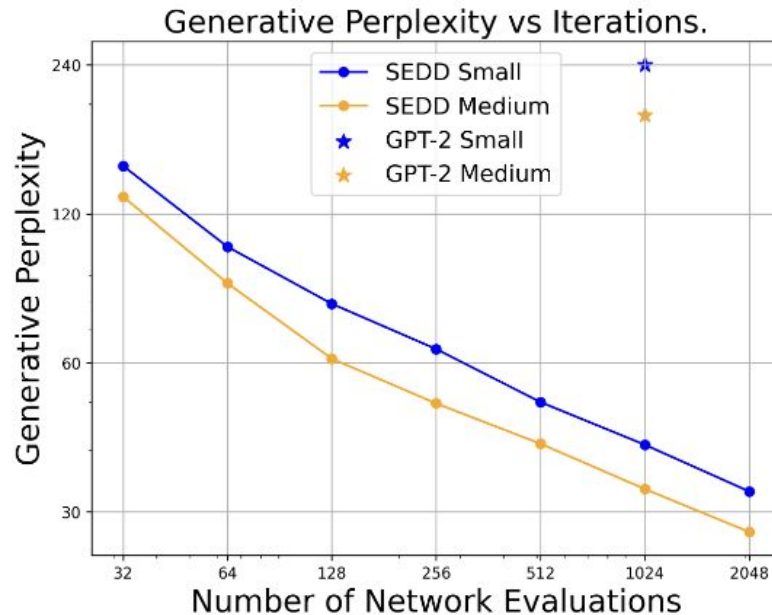
“If they’ve requested to do it and they’re still not doing it, don’t consider there an artificial interest here,” said Flavio Witeli, an agency lawyer, who focuses in cybersecurity law.

To help with Do and Co’s troubles, employees find themselves retraining from software products.



# Putting it all together

GPT-2 S	a hiring platform that "includes a fun club meeting place," says petitioner's AQQFredricks. They's the adjacent marijuana-hop. Others have allowed 3B Entertainment
GPT-2 M	misused, whether via Uber, a higher-order reality of quantified impulse or the No Mass Paralysis movement, but the most shamefully universal example is gridlock
SEDD S	As Jeff Romer recently wrote, "The economy has now reached a corner - 64% of household wealth and 80% of wealth goes to credit cards because of government austerity
SEDD M	Wyman worked as a computer science coach before going to work with the U.S. Secret Service in upstate New York in 2010. Without a license, the Secret Service will have to



Surpasses autoregressive transformers for generation quality/speed!

# Conditional Sampling

$a \square \square d$

$$\frac{p(bc|ad)}{p(bb|ad)} = \frac{p(abcd)/p(ad)}{p(abbd)/p(ad)} = \frac{p(abcd)}{p(abbd)}$$

$S_\theta$  can be reused (just don't change the filled-in indices)

# Conditional Generation (Prompt Infilling)

A bow and arrow is a traditional weapon that enables an attacker to attack targets at a range within a meter or maybe two meters. They have a range far longer than a human can walk, and they can be fired ...

... skydiving is a fun sport that makes me feel incredibly silly. I think I may've spent too much, but it could've been amazing! While sky diving gives us exercise and fun, scuba diving is an act of physical fitness, ...

... no one expected the results to much better than last year's one-sided endorsement. Nearly 90 percent of the results were surveyed as "independent," an promising result for school children across the country.

... results show that Donald Trump and Hillary Clinton are in 38 states combined with less than 1% of the national vote. In a way, it's Trump and Hillary Clinton who will work overtime to get people to vote this ...

Method	Annealing	Mauve ( $\uparrow$ )
GPT-2	Nucleus-0.95	0.955
	None	0.802
SEDD Standard	None	<b>0.957</b>
SEDD Infill	None	0.942

Matches best GPT-2 quality without hacks and with general prompts!

# Computing Likelihood Bounds

$$-\log p_{\theta}(x_0) \leq \int_0^t \mathbb{E}_{x_t \sim p_t(\cdot|x_0)} \sum_{y \neq x_t} Q_t(x_t, y) \left( s_{\theta}(x_t, t)_y - \frac{p_{t|0}(y|x_0)}{p_{t|0}(x_t|x_0)} \log s_{\theta}(x_t, t)_y \right) dt + C$$

(Weighted) version of score entropy.

$$PPL(x) \leq e^{-\frac{1}{d} DSE(x)}$$

# Computing Likelihood Bounds

Size	Model	LAMBADA	WikiText2	PTB	WikiText103	1BW
Small	GPT-2	<b>45.04</b>	42.43	138.43	41.60	<b>75.20*</b>
	SEDD Absorb	$\leq 50.92$	$\leq \mathbf{41.84}$	$\leq \mathbf{114.24}$	$\leq \mathbf{40.62}$	$\leq 79.29$
	SEDD Uniform	$\leq 65.40$	$\leq 50.27$	$\leq 140.12$	$\leq 49.60$	$\leq 101.37$
Medium	GPT-2	<b>35.66</b>	31.80	123.14	31.39	<b>55.72*</b>
	SEDD Absorb	$\leq 42.77$	$\leq \mathbf{31.04}$	$\leq \mathbf{87.12}$	$\leq \mathbf{29.98}$	$\leq 61.19$
	SEDD Uniform	$\leq 51.28$	$\leq 38.93$	$\leq 102.28$	$\leq 36.81$	$\leq 79.12$

Challenges autoregressive modeling on perplexities!

# Summary

- It is hard to build probabilistic models for discrete space.
  - Autoregressive modeling has been (basically) the only paradigm
- Concrete score based models
  - Model the ratios of the data distribution (concrete scores)
  - Optimize Score Entropy loss (+ extensions)
- Sample using discrete diffusion processes
  - Synergizes with Denoising Score Entropy loss
  - Fast and controllable generation
  - Generation quality surpasses autoregressive models
- Score Entropy forms a likelihood bound.
  - Challenges autoregressive dominance