Parallel sampling via autospeculation

Thuy-Duong "June" Vuong

UC San Diego

Simons Modern Paradigms in Generalization
Reunion Workshop, January 20-23, 2026

Based on joint work with
Nima Anari, Carlo Baronio, CJ Chen, Alireza Haqi, Frederic Koehler and Anqi Li

Generative modeling

Autoregression (ChatGPT,…)

- $\mu : [q]^n \to \mathbb{R}_{\geq 0}$
- Sample via BERT-style marginal oracle

$$\mu(X_i = x_i | X_S = x_S)$$

_____ _____ _____ kitchen today.

renovate	10%
clean	45%
use	45%

Generative modeling

Autoregression (ChatGPT,…)

- $\mu : [q]^n \to \mathbb{R}_{\geq 0}$
- Sample via conditional marginal oracle

$$\mu(X_i = x_i | X_S = x_S)$$

- Sample by any-order autoregression

Generative modeling

Autoregression (ChatGPT,…)

- $\mu : [q]^n \to \mathbb{R}_{\geq 0}$
- Sample via conditional marginal oracle

$$\mu(X_i = x_i | X_S = x_S)$$

$$\_\_\_\_\_ \ \_\_\_\_\_ \ \_\_\_\_\_ \ \_\_\_\_\_ \ \_\_\_\_\_.$$

Denoising diffusion (DALL-E,…)

- $\mu : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$
- Sample via conditional mean oracle

$$f(t, x) = \mathbb{E}_{Y \sim \mu, g \sim \mathcal{N}(0, \frac{1}{t} \cdot I)} [Y | Y + g = \frac{x}{t}]$$

Generative modeling

Autoregression (ChatGPT,…)

- $\mu : [q]^n \to \mathbb{R}_{\geq 0}$
- Sample via conditional marginal oracle
$$\mu(X_i = x_i | X_S = x_S)$$

_____ _____ _____ _____ _____.

today	15%
Tuesday	5%
never	20%
….	

Generative modeling

Autoregression (ChatGPT,…)

- $\mu : [q]^n \to \mathbb{R}_{\geq 0}$
- Sample via conditional marginal oracle

$$\mu(X_i = x_i | X_S = x_S)$$

_____ _____ _____ _____ today.

Generative modeling

Autoregression (ChatGPT,…)

- $\mu : [q]^n \to \mathbb{R}_{\geq 0}$
- Sample via conditional marginal oracle
$$\mu(X_i = x_i | X_S = x_S)$$

Tom _____ _____ _____ today.

Generative modeling

Autoregression (ChatGPT,…)

- $\mu : [q]^n \to \mathbb{R}_{\geq 0}$
- Sample via conditional marginal oracle

$$\mu(X_i = x_i | X_S = x_S)$$

Tom will _____ _____ today.

Generative modeling

Autoregression (ChatGPT,…)

- $\mu : [q]^n \to \mathbb{R}_{\geq 0}$
- Sample via conditional marginal oracle

$$\mu(X_i = x_i | X_S = x_S)$$

Tom will eat ______ today.

Generative modeling

Autoregression (ChatGPT,…)

- $\mu : [q]^n \to \mathbb{R}_{\geq 0}$
- Sample via conditional marginal oracle
$$\mu(X_i = x_i | X_S = x_S)$$

Tom will eat breakfast today.

Generative modeling

Autoregression (ChatGPT,…)

- $\mu : [q]^n \to \mathbb{R}_{\geq 0}$
- Sample via conditional marginal oracle

$$\mu(X_i = x_i | X_S = x_S)$$

- Sample by any-order autoregression

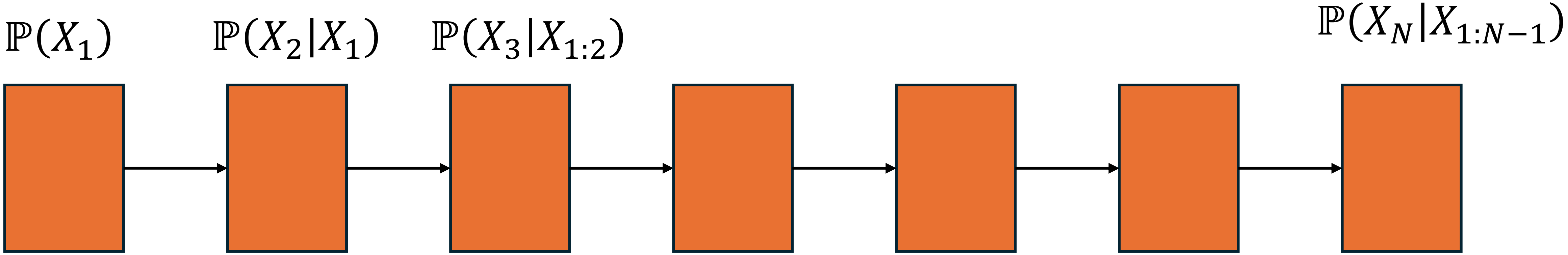
Denoising diffusion (DALL-E,…)

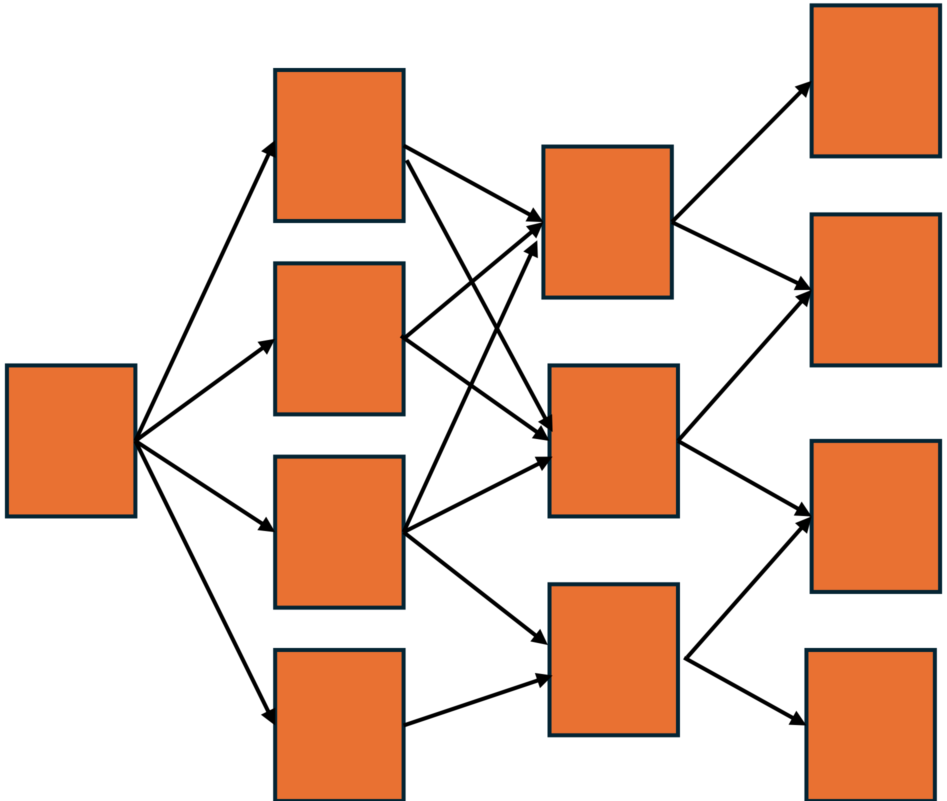
- $\mu : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$
- Sample via conditional mean oracle

$$f(t, x) = \mathbb{E}_{Y\sim\mu, g\sim\mathcal{N}(0, \frac{1}{t}\cdot I)} [Y | Y + g = \frac{x}{t}]$$

- Sample by stochastic differential eq (SDE)

Parallelization?





- Seems inherently sequential. Parallelization?
- Parallel computing model:
  - Many parallel processes per round
  - Communication at the end of each round
- Our results: autoregression (& diffusion)

  in $\tilde{O}(n^{\frac{1}{2}})$ rounds

  and $O(n \log n)$ total work

Autoregression (ChatGPT,…)

General $\mu: [q]^n \to \mathbb{R}_{\geq 0}$, BERT marginals $\mu(X_i = x_i | X_S = x_S)$

- Folklore: $\Theta(n)$ oracle queries/total work
- [AGR24] :
  - Upper bound: $\tilde{O}(n^{\frac{2}{3}})$ rounds & $O(n \log n)$ queries/total work
  - Lower bound: $\Omega(n^{\frac{1}{3}})$ rounds $\forall$ poly-queries algorithm
- This work:
  - $\tilde{O}(n^{\frac{1}{2}})$ rounds & $O(n \log n)$ queries/total work
  - Simpler algorithm that also works for diffusion

Autoregression (ChatGPT,…)

General $\mu: [q]^n \to \mathbb{R}_{\geq 0}$, BERT marginals $\mu(X_i = x_i | X_S = x_S)$

- Folklore: $\Theta(n)$ oracle queries/total work
- [AGR24] :
  - Upper bound: $\tilde{O}(n^{\frac{2}{3}})$ rounds & $O(n \log n)$ queries/total work
  - Lower bound: $\Omega(n^{\frac{1}{3}})$ rounds ∀ poly-queries algorithm
- This work:
  - $\tilde{O}(n^{\frac{1}{2}})$ rounds & $O(n \log n)$ queries/total work

- BERT marginals are necessary for parallelization:
  - Can't get $o(n)$ parallel time with only fixed-order marginals $\mu(X_i | X_{1:i-1})$

Algorithm

Rejection sampling

μ: target distribution

ν: reference distribution

The following exactly samples from μ:

1. Sample $\boldsymbol{x} \sim \nu$
2. Accept and return $\boldsymbol{x}$ with probability $\frac{\mu}{\nu}(\boldsymbol{x})$

$$\rightarrow \mathbb{P}[\text{output x}] = \nu(\boldsymbol{x}) \cdot \frac{\mu}{\nu}(\boldsymbol{x}) = \mu(x)$$

Rejection sampling

μ: target distribution

ν: reference distribution

The following exactly samples from μ:

1. Sample $\boldsymbol{x} \sim \nu$
2. Accept and return $\boldsymbol{x}$ with probability $\frac{\mu}{\nu}(\boldsymbol{x})$ What if $\frac{\mu}{\nu}(\boldsymbol{x}) > 1$?

$$\rightarrow \mathbb{P}[\text{output x}] = \nu(\boldsymbol{x}) \cdot \frac{\mu}{\nu}(\boldsymbol{x}) = \mu(x)$$

Speculative rejection sampling

μ: target distribution

ν: reference distribution

The following exactly samples from μ:

1. Sample $\boldsymbol{x} \sim \nu$
2. Accept and return $\boldsymbol{x}$ with probability $\min\{1, \frac{\mu}{\nu}(\boldsymbol{x})\}$

$$\rightarrow \mathbb{P}[\text{output x}] = \min\{\mu(x), \nu(x)\}$$

Wrong output distribution!

Speculative rejection sampling

μ: target distribution

ν: reference distribution

The following exactly samples from μ:

1. Sample $\boldsymbol{x} \sim \nu$
2. Accept and return $\boldsymbol{x}$ with probability $\min\{1, \frac{\mu}{\nu}(\boldsymbol{x})\}$

   $\rightarrow \mathbb{P}[\text{output x}] = \min\{\mu(x), \nu(x)\}$
3. While not accepted do:
   Sample $\boldsymbol{y} \sim \mu$
   Accept and return $\boldsymbol{y}$ with probability $\max\{0, 1 - \frac{\nu}{\mu}(\boldsymbol{y})\}$

$\rightarrow \mathbb{P}[\text{output y}] = \max\{0, \mu(y) - \nu(y)\}$

Recursive speculative rejection sampling

μ: target distribution

ν: reference distribution

The following exactly samples from μ: How to implement?

1. Sample $\boldsymbol{x} \sim \nu$
2. Accept and return $\boldsymbol{x}$ with probability $\min\{1, \frac{\mu}{\nu}(\boldsymbol{x})\}$
3. While not accepted do:
   Sample $\boldsymbol{y} \sim \mu$ by: $y_{1:n/2} \sim \mu(X_{1:n/2}), y_{n/2+1:n} \sim \mu(X_{n/2+1:n} | X_{1:n/2} = y_{1:n/2})$
   Accept and return $\boldsymbol{y}$ with probability $\max\{0, 1 - \frac{\nu}{\mu}(\boldsymbol{y})\}$

Recursive speculative rejection sampling

μ: target distribution

ν: reference distribution that admits fast sampler e.g. in $O(1)$ rounds
& can be built w/ conditional marginal/mean oracles for μ

The following exactly samples from μ: How to implement?

1. Sample $\boldsymbol{x} \sim \nu$
2. Accept and return $\boldsymbol{x}$ with probability $\min\{1, \frac{\mu}{\nu}(\boldsymbol{x})\}$
3. While not accepted do:

   ...

- Fully sequential: no gain
- Fully parallel: need infinite #processors

Recursive speculative rejection sampling

μ: target distribution

ν: reference distribution

The following exactly samples from μ:

How to implement?

1. Sample $\boldsymbol{x} \sim \nu$
2. Accept and return $\boldsymbol{x}$ with probability $\min\{1, \frac{\mu}{\nu}(\boldsymbol{x})\}$
3. While not accepted do:

   ...

- Fully sequential: no gain
- Fully parallel: need infinite #processors
- Key idea: sequentially process batches of geometrically increasing size, where each batch is processed in parallel

Recursive speculative rejection sampling

μ: target distribution

ν: reference distribution

The following exactly samples from μ:

1. Sample $\boldsymbol{x} \sim \nu$
2. Accept and return $\boldsymbol{x}$ with probability $\min\{1, \frac{\mu}{\nu}(\boldsymbol{x})\}$
3. For r=0,1,2,... do:

   For $i = \lceil (1+\rho)^r \rceil, \lceil (1+\rho)^r \rceil + 1, \cdots, \lceil (1+\rho)^{r+1} \rceil - 1$ do in parallel
   Sample $\boldsymbol{y} \sim \mu$ by:

   $$y_{1:n/2} \sim \mu\left(X_{1:n/2}\right), y_{n/2+1:n} \sim \mu(X_{n/2+1:n} | X_{1:n/2} = y_{1:n/2})$$

   Accept and return $\boldsymbol{y}$ with probability $\max\{0, 1 - \frac{\nu}{\mu}(\boldsymbol{y})\}$

Recursive speculative rejection sampling

- $S = \{a+1, \cdots, b\}$
- μ_S: target distribution
- ν_S: reference distribution

In O(1) rounds and O(|S|) queries, can:

- Compute $\frac{d\mu_S}{d\nu_S}$
- Sample ν_S

The following exactly samples from μ_S:

1. Sample $\boldsymbol{x} \sim \nu_S$
2. Accept and return $\boldsymbol{x}$ with probability $\min\{1, \frac{\mu_S}{\nu_S}(\boldsymbol{x})\}$
3. For r=0,1,2,... do:

   For $i = \lceil (1+\rho)^r \rceil, \lceil (1+\rho)^r \rceil + 1, \cdots, \lceil (1+\rho)^{r+1} \rceil - 1$ do in parallel:

   Sample $\boldsymbol{y} \sim \mu_S$ by: Set $\mathrm{L}(S) = \left\{a+1, \dots, \frac{a+b}{2}\right\}, \mathrm{R}(S) = \{\frac{a+b}{2} + 1, \dots, b\}$

   $$y_{L(S)} \sim \mu_{L(S)}, y_{R(S)} \sim \mu_{R(S)}(\cdot \mid X_{<R(S)} = ..|| y_{L(S)})$$

   Accept and return $\boldsymbol{y}$ with probability $\max\{0, 1 - \frac{\nu_S}{\mu_S}(\boldsymbol{y})\}$

What are μ_S and ν_S?

- $\mu : [q]^n \to \mathbb{R}_{\geq 0}$
- Oracle access to BERT-style marginals $\mu(X_i = x_i | X_S = x_S)$
- Sample order $\sigma \sim \mathrm{Uniform}(S_n)$
  - $S = \{a+1, .., b\}$
  - $\mu_S = \mu\left(X_{\sigma(S)} \middle| X_{\sigma(<S)}\right) = \mu(X_{\sigma(a+1:b)} | X_{\sigma(1:a)})$
  - $\nu_S =$ product distribution with same marginal as μ_S

    $= \mu\left(X_{\sigma(a+1)} \middle| X_{\sigma(1:a)}\right) \otimes \cdots \otimes \mu\left(X_{\sigma(b)} \middle| X_{\sigma(1:a)}\right)$

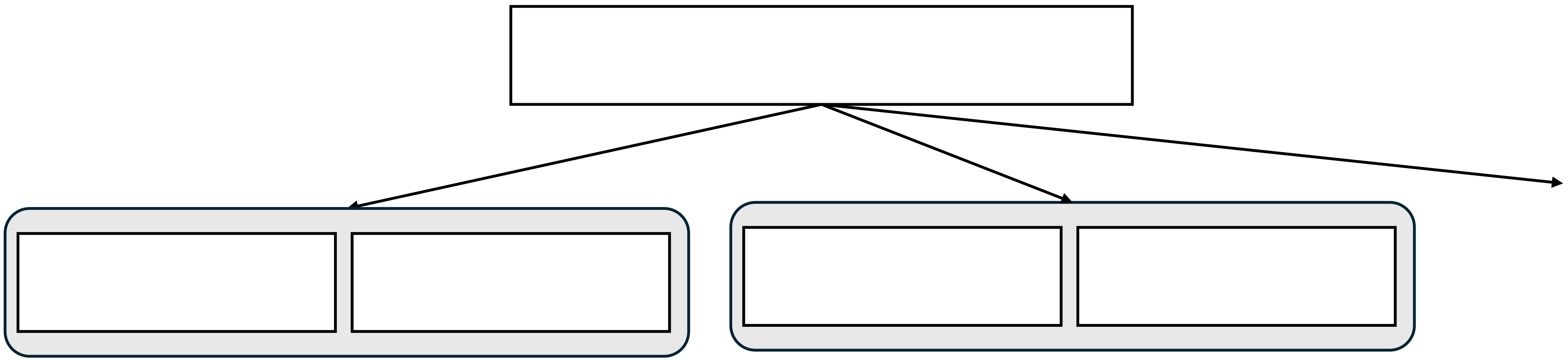
  Can sample each coordinate of ν_S in parallel b/c no dependencies

For fixed order σ, $TV(\mu_S, \nu_S) = \Omega(1)$ b/c dependencies

Pinning lemma: $\mathbb{E}_{\sigma \sim S_n}[TV(\mu_S, \nu_S)] \approx |S|/\sqrt{n}$

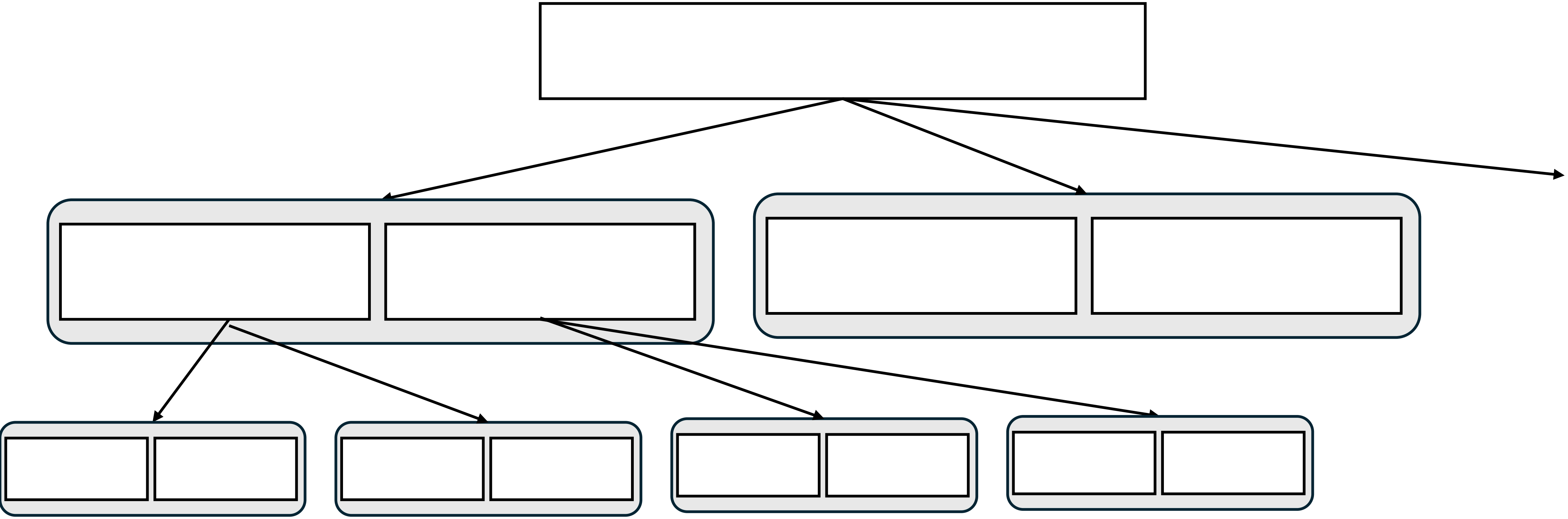
Analysis: high level

- Difficulty: Many parallel threads & deep recursion



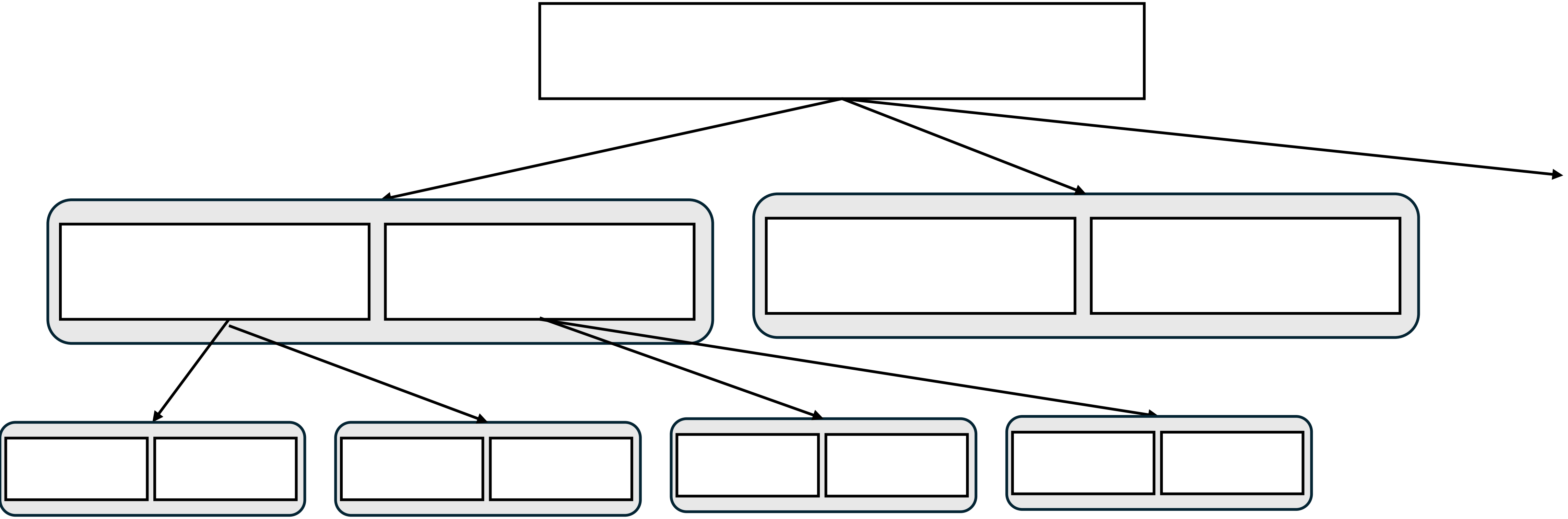
Analysis: high level

- Difficulty: Many parallel threads & deep recursion



Analysis: high level

- Difficulty: Many parallel threads & deep recursion
- $\mathbb{E}[T] = \mathbb{E}[\sum_S TV(\mu_S, \nu_S)] \approx \tilde{O}\left(\sqrt{n}\right)$ Pinning lemma



Summary & open questions

General $\mu: [q]^n \to \mathbb{R}_{\geq 0}$, BERT marginals $\mu(X_i = x_i | X_S = x_S)$

- [AGR24] :
  - Upper bound: $\tilde{O}(n^{\frac{2}{3}})$ rounds & $O(n \log n)$ queries/total work
  - Lower bound: $\Omega(n^{\frac{1}{3}})$ rounds $\forall$ poly-queries algorithm
- This work:
  - $\tilde{O}(n^{\frac{1}{2}})$ rounds & $O(n \log n)$ queries/total work
  - Simpler algorithm that also works for diffusion
- Open:
  - Handling more noise via (parallelized?) back-tracking