Deep Generative Models

Lecture 14

Roman Isachenko

AI Masters

2026, Spring

Recap of Previous Lecture

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}(0))}\mathbb{E}_{t\sim U[0,1]}\mathbb{E}_{q(\mathbf{x}(t)|\mathbf{x}(0))}\left\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}(t),t)-\nabla_{\mathbf{x}(t)}\log q(\mathbf{x}(t)|\mathbf{x}(0))\right\|_2^2$$

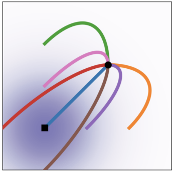
$$p_t(\mathbf{x}|\mathbf{x}_1) = q_{1-t}(\mathbf{x}|\mathbf{x}_0 = \mathbf{x}_1)$$

Variance Exploding SDE

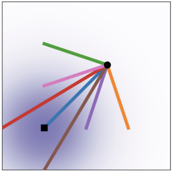
$$p_t(\mathbf{x}|\mathbf{x}_1) = \mathcal{N}\left(\mathbf{x}_1, \sigma_{1-t}^2\mathbf{I}\right) \quad \Rightarrow \quad \mathbf{f}(\mathbf{x}, \mathbf{x}_1, t) = -\frac{\sigma'_{1-t}}{\sigma_{1-t}}(\mathbf{x}_t - \mathbf{x}_1)$$

Variance Preserving SDE

$$p_t(\mathbf{x}_t|\mathbf{x}_1) = \mathcal{N}\left(\alpha_{1-t}\mathbf{x}_1, (1-\alpha_{1-t}^2)\mathbf{I}\right) \Rightarrow \mathbf{f}(\mathbf{x}_t, \mathbf{x}_1, t) = \frac{\alpha'_{1-t}}{1-\alpha_{1-t}^2}\cdot(\alpha_{1-t}\mathbf{x}_t - \mathbf{x}_1)$$



Diffusion



OT

Lipman Y., et al. Flow Matching for Generative Modeling, 2022

Recap of Previous Lecture

Continuous state space

- **Discrete time** $t \in \{0, 1, \ldots, T\}$ $\Rightarrow$ **DDPM / NCSN**.
- **Continuous time** $t \in [0, 1]$ $\Rightarrow$ **Score-based SDE models**.

Discrete state space

- **Discrete time** $t \in \{0, 1, \ldots, T\}$.
- **Continuous time** $t \in [0, 1]$.

Key advantages of discrete diffusion

- Parallel generation
- Flexible infilling
- Robustness
- Unified framework

Recap of Previous Lecture

Discrete Diffusion Markov Chain

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathrm{Cat}(\mathbf{Q}_t\mathbf{x}_{t-1}),$$

Each $\mathbf{x}_t \in \{0,1\}^K$ is a **one-hot vector** encoding the categorical state (it is just one token).

Transition Matrix

$$[\mathbf{Q}_t]_{ij} = q(x_t = i|x_{t-1} = j), \qquad \sum_{i=1}^{K}[\mathbf{Q}_t]_{ij} = 1.$$

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathrm{Cat}(\mathbf{Q}_{1:t}\mathbf{x}_0), \qquad \mathbf{Q}_{1:t} = \mathbf{Q}_t\mathbf{Q}_{t-1}\cdots\mathbf{Q}_1.$$

- The choice of $\mathbf{Q}_t$ determines how information is erased and what the stationary distribution becomes.
- $\mathbf{Q}_t$ and $\mathbf{Q}_{1:t}$ should be easy to compute for each t.

Austin J. et al. Structured denoising diffusion models in discrete state-spaces, 2021.

Recap of Previous Lecture

Uniform vs. Absorbing Transition Matrix

Aspect	**Uniform Diffusion**	**Absorbing Diffusion**
$\mathbf{Q}_t$	$(1-\beta_t)\mathbf{I} + \beta_t\mathbf{U}$	$(1-\beta_t)\mathbf{I} + \beta_t\mathbf{e}_m\mathbf{1}^\top$
$\mathbf{Q}_{1:t}$	$\bar{\alpha}_t\mathbf{I} + (1-\bar{\alpha}_t)\mathbf{U}$	$\bar{\alpha}_t\mathbf{I} + (1-\bar{\alpha}_t)\mathbf{e}_m\mathbf{1}^\top$
$\mathbf{Q}_{1:\infty}$	$\mathbf{U}$	$\mathrm{Cat}(\mathbf{e}_m)$
Interpretation	Random replacement	Gradual masking of tokens
Application	Image diffusion	Text diffusion $\approx$ Masked LM

Observation

Both schemes gradually destroy information, but differ in their stationary limit. Absorbing diffusion bridges diffusion and masked-language-model objectives.

Austin J. et al., Structured denoising diffusion models in discrete state-spaces, 2021.

Recap of Previous Lecture

ELBO

$$\mathcal{L}_{\phi,\theta}(\mathbf{x}) = \mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} \log p_\theta(\mathbf{x}_0|\mathbf{x}_1) - \mathrm{KL}\big(q(\mathbf{x}_T|\mathbf{x}_0)\|p(\mathbf{x}_T)\big) - \\ - \sum_{t=2}^{T} \underbrace{\mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} \mathrm{KL}\big(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)\|p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)\big)}_{\mathcal{L}_t}$$

Discrete conditioned reverse distribution

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathrm{Cat}\left(\frac{\mathbf{Q}_t \mathbf{x}_t \odot \mathbf{Q}_{1:t-1}\mathbf{x}_0}{\mathbf{x}_t^\top \mathbf{Q}_{1:t}\mathbf{x}_0}\right).$$

- Both $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ and $q(\mathbf{x}_t|\mathbf{x}_0)$ are known analytically from the forward process.
- The reverse process $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is a learned categorical distribution:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathrm{Cat}\big(\boldsymbol{\pi}_\theta(\mathbf{x}_t, t)\big).$$

Austin J. et al., Structured denoising diffusion models in discrete state-spaces, 2021.

Recap of Previous Lecture

ELBO term

$$\mathcal{L}_t = \mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} \mathrm{KL}\big(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \,\|\, p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1}|\mathbf{x}_t)\big).$$

Categorical KL

$$\mathrm{KL}\big(\mathrm{Cat}(\mathbf{q}) \,\|\, \mathrm{Cat}(\mathbf{p})\big) = \sum_{k=1}^{K} q_k \log \frac{q_k}{p_k} = \mathrm{H}(\mathbf{q}, \mathbf{p}) - \mathrm{H}(\mathbf{q}),$$

- $\mathrm{H}\big(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)\big)$ is a constant w.r.t. $\boldsymbol{\theta}$.
- $\mathrm{H}(\mathbf{q}, \mathbf{p}) = -\sum_k q_k \log p_k$ is a **cross-entropy loss**.

Therefore, minimizing $\mathcal{L}_t$ w.r.t. $\boldsymbol{\theta}$ is equivalent to minimizing

$$\mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} \mathrm{H}\Big(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0),\; p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1}|\mathbf{x}_t)\Big).$$

Austin J. et al., Structured denoising diffusion models in discrete state-spaces, 2021.

Outline

Outline

1. Discrete Diffusion
   From Token to Sequence
   Absorbing Diffusion
   Continuous-Time Formulation

2. Course Overview

Outline

From Token to Sequence

One-hot sequence representation

$$\mathbf{x}_t \in \{0,1\}^K \quad \Leftrightarrow \quad \mathbf{X}_t \in \{0,1\}^{K \times m}$$

Here $\mathbf{X}_t$ is a one-hot representation of a sequence of tokens.

From Token to Sequence

One-hot sequence representation

$$\mathbf{x}_t \in \{0,1\}^K \quad \Leftrightarrow \quad \mathbf{X}_t \in \{0,1\}^{K \times m}$$

Here $\mathbf{X}_t$ is a one-hot representation of a sequence of tokens.

Independent Token-wise Forward Process

$$q(\mathbf{X}_t | \mathbf{X}_{t-1}) = \prod_{i=1}^{m} q(\mathbf{x}_t^i | \mathbf{x}_{t-1}^i) = \mathrm{Cat}(\mathbf{Q}_t \mathbf{X}_{t-1})$$

- Each position i evolves according to its own Markov chain.
- Often the same transition matrix $\mathbf{Q}_t$ is shared across i.

From Token to Sequence

One-hot sequence representation

$$\mathbf{x}_t \in \{0,1\}^K \quad \Leftrightarrow \quad \mathbf{X}_t \in \{0,1\}^{K\times m}$$

Here $\mathbf{X}_t$ is a one-hot representation of a sequence of tokens.

Independent Token-wise Forward Process

$$q(\mathbf{X}_t|\mathbf{X}_{t-1}) = \prod_{i=1}^{m} q(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i) = \mathrm{Cat}(\mathbf{Q}_t\mathbf{X}_{t-1})$$

- Each position i evolves according to its own Markov chain.
- Often the same transition matrix $\mathbf{Q}_t$ is shared across i.

Continuous Diffusion Analogy

- In Gaussian DDPMs with diagonal covariance, noise is independent per pixel.
- Structure is not in the noise; it is learned by the reverse model.

From Token to Sequence

$$q(\mathbf{X}_t|\mathbf{X}_{t-1}) = \prod_{i=1}^{m} q(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i) = \mathrm{Cat}(\mathbf{Q}_t\mathbf{X}_{t-1})$$

From Token to Sequence

$$q(\mathbf{X}_t|\mathbf{X}_{t-1}) = \prod_{i=1}^{m} q(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i) = \mathrm{Cat}(\mathbf{Q}_t\mathbf{X}_{t-1})$$

$$q(\mathbf{X}_t|\mathbf{X}_0) = \mathrm{Cat}(\mathbf{Q}_{1:t}\mathbf{X}_0)$$

From Token to Sequence

$$q(\mathbf{X}_t|\mathbf{X}_{t-1}) = \prod_{i=1}^{m} q(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i) = \mathrm{Cat}(\mathbf{Q}_t\mathbf{X}_{t-1})$$

$$q(\mathbf{X}_t|\mathbf{X}_0) = \mathrm{Cat}(\mathbf{Q}_{1:t}\mathbf{X}_0)$$

Conditioned Reverse Distribution

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathrm{Cat}\left(\frac{\mathbf{Q}_t\mathbf{x}_t \odot \mathbf{Q}_{1:t-1}\mathbf{x}_0}{\mathbf{x}_t^\top \mathbf{Q}_{1:t}\mathbf{x}_0}\right).$$

$$q(\mathbf{X}_{t-1}|\mathbf{X}_t, \mathbf{X}_0) = \prod_{i=1}^{m} q(\mathbf{x}_{t-1}^i|\mathbf{x}_t^i, \mathbf{x}_0^i).$$

From Token to Sequence

$$q(\mathbf{X}_t|\mathbf{X}_{t-1}) = \prod_{i=1}^{m} q(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i) = \mathrm{Cat}(\mathbf{Q}_t\mathbf{X}_{t-1})$$

$$q(\mathbf{X}_t|\mathbf{X}_0) = \mathrm{Cat}(\mathbf{Q}_{1:t}\mathbf{X}_0)$$

Conditioned Reverse Distribution

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathrm{Cat}\left(\frac{\mathbf{Q}_t\mathbf{x}_t \odot \mathbf{Q}_{1:t-1}\mathbf{x}_0}{\mathbf{x}_t^\top \mathbf{Q}_{1:t}\mathbf{x}_0}\right).$$

$$q(\mathbf{X}_{t-1}|\mathbf{X}_t, \mathbf{X}_0) = \prod_{i=1}^{m} q(\mathbf{x}_{t-1}^i|\mathbf{x}_t^i, \mathbf{x}_0^i).$$

- All distributions defined by the forward process are factorized.
- Dependence appears in the learned reverse model.

Reverse Model for Sequence

$$p_\theta(\mathbf{X}_{t-1}|\mathbf{X}_t) = \prod_{i=1}^{m} p_\theta(\mathbf{x}_{t-1}^i|\mathbf{X}_t).$$

- The output factorizes (parallel prediction across positions).
- Each factor conditions on the entire noisy sequence $\mathbf{X}_t$.
- This is exactly the **masked language modeling** pattern.

Reverse Model for Sequence

$$p_\theta(\mathbf{X}_{t-1}|\mathbf{X}_t) = \prod_{i=1}^{m} p_\theta(\mathbf{x}_{t-1}^i|\mathbf{X}_t).$$

- The output factorizes (parallel prediction across positions).
- Each factor conditions on the entire noisy sequence $\mathbf{X}_t$.
- This is exactly the **masked language modeling** pattern.

Objective: $\mathcal{L}_t$ term

$$\begin{aligned}\mathrm{KL}\left(q(\mathbf{X}_{t-1}|\mathbf{X}_t, \mathbf{X}_0) \parallel p_\theta(\mathbf{X}_{t-1}|\mathbf{X}_t)\right) \\ = \sum_{i=1}^{m} \mathrm{KL}\left(q(\mathbf{x}_{t-1}^i|\mathbf{x}_t^i, \mathbf{x}_0^i) || p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1}^i|\mathbf{X}_t)\right).\end{aligned}$$

Reverse Model for Sequence

$$p_\theta(\mathbf{X}_{t-1}|\mathbf{X}_t) = \prod_{i=1}^{m} p_\theta(\mathbf{x}^i_{t-1}|\mathbf{X}_t).$$

- The output factorizes (parallel prediction across positions).
- Each factor conditions on the entire noisy sequence $\mathbf{X}_t$.
- This is exactly the **masked language modeling** pattern.

Objective: $\mathcal{L}_t$ term

$$\begin{aligned}\mathrm{KL}\left(q(\mathbf{X}_{t-1}|\mathbf{X}_t, \mathbf{X}_0) \,\|\, p_\theta(\mathbf{X}_{t-1}|\mathbf{X}_t)\right) \\ = \sum_{i=1}^{m} \mathrm{KL}\left(q(\mathbf{x}^i_{t-1}|\mathbf{x}^i_t, \mathbf{x}^i_0)||p_{\boldsymbol{\theta}}(\mathbf{x}^i_{t-1}|\mathbf{X}_t)\right).\end{aligned}$$

Final objective: masked LM

$$\mathcal{L} = \sum_{t=1}^{T} \sum_{i=1}^{m} \mathbb{E}_{q(\mathbf{X}_t|\mathbf{X}_0)}\Big[-\log p_{\boldsymbol{\theta}}(\mathbf{x}^i_0|\mathbf{X}_t)\Big].$$

Outline

Absorbing Diffusion: Forward Process

Let's restrict to the case of absorbing transition matrix.

$$\mathbf{Q}_t = (1-\beta_t)\mathbf{I} + \beta_t\,\mathbf{e}_m\mathbf{1}^\top, \qquad \bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s).$$

$$\mathbf{Q}_{1:t} = \bar{\alpha}_t\,\mathbf{I} + (1-\bar{\alpha}_t)\,\mathbf{e}_m\mathbf{1}^\top.$$

Sahoo S. et al. Simple and effective masked diffusion language models, 2024

Absorbing Diffusion: Forward Process

Let's restrict to the case of absorbing transition matrix.

$$\mathbf{Q}_t = (1-\beta_t)\mathbf{I} + \beta_t\,\mathbf{e}_m\mathbf{1}^\top, \qquad \bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s).$$

$$\mathbf{Q}_{1:t} = \bar{\alpha}_t\,\mathbf{I} + (1-\bar{\alpha}_t)\,\mathbf{e}_m\mathbf{1}^\top.$$

Each position is either still clean or already masked:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \bar{\alpha}_t[\mathbf{x}_t = \mathbf{x}_0] + (1-\bar{\alpha}_t)[\mathbf{x}_t = \mathbf{e}_m]$$

Sahoo S. et al. Simple and effective masked diffusion language models, 2024

Absorbing Diffusion: Forward Process

Let's restrict to the case of absorbing transition matrix.

$$\mathbf{Q}_t = (1-\beta_t)\mathbf{I} + \beta_t\,\mathbf{e}_m \mathbf{1}^\top, \qquad \bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s).$$

$$\mathbf{Q}_{1:t} = \bar{\alpha}_t\,\mathbf{I} + (1-\bar{\alpha}_t)\,\mathbf{e}_m \mathbf{1}^\top.$$

Each position is either still clean or already masked:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \bar{\alpha}_t[\mathbf{x}_t = \mathbf{x}_0] + (1-\bar{\alpha}_t)[\mathbf{x}_t = \mathbf{e}_m]$$

$$\mathbf{Q}_t = \begin{pmatrix} 1-\beta_t & 0 & 0 \\ 0 & 1-\beta_t & 0 \\ \beta_t & \beta_t & 1 \end{pmatrix} \quad \Rightarrow \quad \text{the masked state is absorbing.}$$

Sahoo S. et al. Simple and effective masked diffusion language models, 2024

Absorbing Diffusion: Forward Process

Let's restrict to the case of absorbing transition matrix.

$$\mathbf{Q}_t = (1 - \beta_t)\mathbf{I} + \beta_t \, \mathbf{e}_m \mathbf{1}^\top, \qquad \bar{\alpha}_t = \prod_{s=1}^{t} (1 - \beta_s).$$

$$\mathbf{Q}_{1:t} = \bar{\alpha}_t \, \mathbf{I} + (1 - \bar{\alpha}_t) \, \mathbf{e}_m \mathbf{1}^\top.$$

Each position is either still clean or already masked:

$$q(\mathbf{x}_t | \mathbf{x}_0) = \bar{\alpha}_t [\mathbf{x}_t = \mathbf{x}_0] + (1 - \bar{\alpha}_t)[\mathbf{x}_t = \mathbf{e}_m]$$

$$\mathbf{Q}_t = \begin{pmatrix} 1 - \beta_t & 0 & 0 \\ 0 & 1 - \beta_t & 0 \\ \beta_t & \beta_t & 1 \end{pmatrix} \quad \Rightarrow \quad \text{the masked state is absorbing.}$$

What happens in the conditioned reverse process $q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)$?

Sahoo S. et al. Simple and effective masked diffusion language models, 2024

Absorbing Diffusion

Conditioned reverse distribution

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \begin{cases} [\mathbf{x}_{t-1} = \mathbf{x}_t], & \text{if } \mathbf{x}_t \neq \mathbf{e}_m, \\ \rho_t[\mathbf{x}_{t-1} = \mathbf{x}_0] + (1 - \rho_t)[\mathbf{x}_{t-1} = \mathbf{e}_m], & \text{if } \mathbf{x}_t = \mathbf{e}_m, \end{cases}$$

where

$$\rho_t = \frac{\beta_t \, \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t}.$$

Absorbing Diffusion

Conditioned reverse distribution

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \begin{cases} [\mathbf{x}_{t-1} = \mathbf{x}_t], & \text{if } \mathbf{x}_t \neq \mathbf{e}_m, \\ \rho_t[\mathbf{x}_{t-1} = \mathbf{x}_0] + (1-\rho_t)[\mathbf{x}_{t-1} = \mathbf{e}_m], & \text{if } \mathbf{x}_t = \mathbf{e}_m, \end{cases}$$

where

$$\rho_t = \frac{\beta_t \, \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t}.$$

- If $\mathbf{x}_t \neq \mathbf{e}_m$, then the token must be unchanged: $\mathbf{x}_{t-1} = \mathbf{x}_t$.
- Observing an unmasked token at time t fixes the entire history: $\mathbf{x}_{t-1} = \mathbf{x}_t = \cdots = \mathbf{x}_0$.

Absorbing Diffusion

Conditioned reverse distribution

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \begin{cases} [\mathbf{x}_{t-1} = \mathbf{x}_t], & \text{if } \mathbf{x}_t \neq \mathbf{e}_m, \\ \rho_t[\mathbf{x}_{t-1} = \mathbf{x}_0] + (1-\rho_t)[\mathbf{x}_{t-1} = \mathbf{e}_m], & \text{if } \mathbf{x}_t = \mathbf{e}_m, \end{cases}$$

where

$$\rho_t = \frac{\beta_t \, \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t}.$$

- If $\mathbf{x}_t \neq \mathbf{e}_m$, then the token must be unchanged: $\mathbf{x}_{t-1} = \mathbf{x}_t$.
- Observing an unmasked token at time t fixes the entire history: $\mathbf{x}_{t-1} = \mathbf{x}_t = \cdots = \mathbf{x}_0$.
- If $\mathbf{x}_t = \mathbf{e}_m$, the previous token may be either clean or masked.
- With probability ρ_t, masking occurred exactly at step t (so $\mathbf{x}_{t-1} = \mathbf{x}_0$).
- With probability $(1 - \rho_t)$, the token was already masked earlier (so $\mathbf{x}_{t-1} = \mathbf{e}_m$).

Absorbing Diffusion

Sequence Distribution

$$q(\mathbf{X}_{t-1}|\mathbf{X}_t, \mathbf{X}_0) = \prod_{i=1}^{m} q(\mathbf{x}_{t-1}^i|\mathbf{x}_t^i, \mathbf{x}_0^i).$$

Each position i has two possible cases:

$$\mathbf{x}_t^i \neq \mathbf{e}_m \;\Rightarrow\; \mathbf{x}_{t-1}^i = \mathbf{x}_t^i, \qquad \mathbf{x}_t^i = \mathbf{e}_m \;\Rightarrow\; \mathbf{x}_{t-1}^i \in \{\mathbf{x}_0^i, \mathbf{e}_m\}.$$

Absorbing Diffusion

Sequence Distribution

$$q(\mathbf{X}_{t-1}|\mathbf{X}_t, \mathbf{X}_0) = \prod_{i=1}^{m} q(\mathbf{x}_{t-1}^i|\mathbf{x}_t^i, \mathbf{x}_0^i).$$

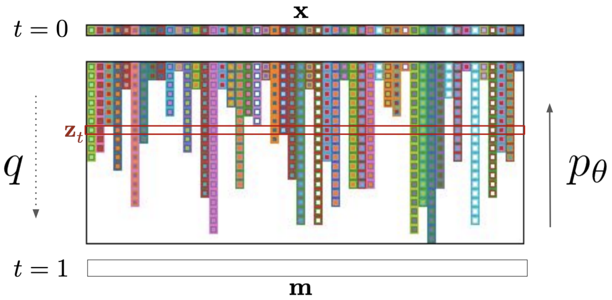
Each position i has two possible cases:

$$\mathbf{x}_t^i \neq \mathbf{e}_m \;\Rightarrow\; \mathbf{x}_{t-1}^i = \mathbf{x}_t^i, \qquad \mathbf{x}_t^i = \mathbf{e}_m \;\Rightarrow\; \mathbf{x}_{t-1}^i \in \{\mathbf{x}_0^i, \mathbf{e}_m\}.$$

Interpretation

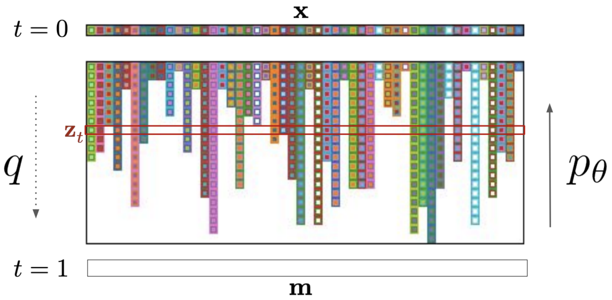
- The forward process produces **random partial observations** of the clean sequence.
- If a token is visible at time t, the reverse distribution is deterministic.
- Unmasked tokens yield a deterministic posterior and therefore contribute only a constant to the ELBO. Therefore, only masked tokens contribute to the training loss.

Absorbing Diffusion



Sahoo S. et al. Simple and effective masked diffusion language models, 2024

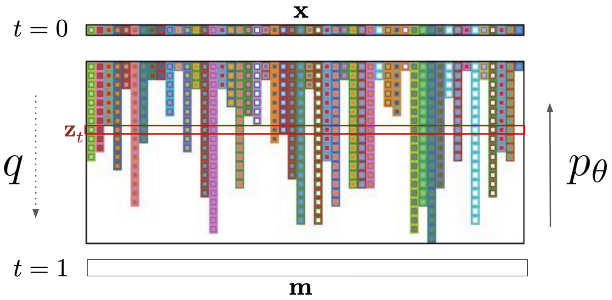
Absorbing Diffusion



$$p_\theta(\mathbf{X}_{t-1}|\mathbf{X}_t) = \prod_{i=1}^{m} p_\theta(\mathbf{x}_{t-1}^i|\mathbf{X}_t).$$

Sahoo S. et al. Simple and effective masked diffusion language models, 2024

Absorbing Diffusion



$$p_\theta(\mathbf{X}_{t-1}|\mathbf{X}_t) = \prod_{i=1}^{m} p_\theta(\mathbf{x}_{t-1}^i|\mathbf{X}_t).$$

Objective: sequence-level $\mathcal{L}_t$

$$\mathcal{L}_t = \mathbb{E}_{q(\mathbf{X}_t|\mathbf{X}_0)} \sum_{i=1}^{m} \rho_t[\mathbf{x}_t^i = \mathbf{e}_m] \left[-\log p_{\boldsymbol{\theta}}(\mathbf{x}_0^i|\mathbf{X}_t)\right] + \text{const}$$

Sahoo S. et al. Simple and effective masked diffusion language models, 2024

Outline

From Discrete Time to Mask Rate

In absorbing diffusion, the forward process is

$$q(\mathbf{x}_t|\mathbf{x}_0) = \bar{\alpha}_t[\mathbf{x}_t = \mathbf{x}_0] + (1 - \bar{\alpha}_t)[\mathbf{x}_t = \mathbf{e}_m].$$

Sahoo S. et al. Simple and effective masked diffusion language models, 2024

From Discrete Time to Mask Rate

In absorbing diffusion, the forward process is

$$q(\mathbf{x}_t|\mathbf{x}_0) = \bar{\alpha}_t[\mathbf{x}_t = \mathbf{x}_0] + (1 - \bar{\alpha}_t)[\mathbf{x}_t = \mathbf{e}_m].$$

- The distribution depends on t only through the scalar

$$\lambda_t = 1 - \bar{\alpha}_t \in [0, 1].$$

Sahoo S. et al. Simple and effective masked diffusion language models, 2024

From Discrete Time to Mask Rate

In absorbing diffusion, the forward process is

$$q(\mathbf{x}_t|\mathbf{x}_0) = \bar{\alpha}_t[\mathbf{x}_t = \mathbf{x}_0] + (1 - \bar{\alpha}_t)[\mathbf{x}_t = \mathbf{e}_m].$$

- The distribution depends on t only through the scalar

$$\lambda_t = 1 - \bar{\alpha}_t \in [0, 1].$$

- We can therefore reparameterize the corruption level by

$$t \quad \Rightarrow \quad \lambda \in [0, 1].$$

Sahoo S. et al. Simple and effective masked diffusion language models, 2024

From Discrete Time to Mask Rate

In absorbing diffusion, the forward process is

$$q(\mathbf{x}_t|\mathbf{x}_0) = \bar{\alpha}_t[\mathbf{x}_t = \mathbf{x}_0] + (1 - \bar{\alpha}_t)[\mathbf{x}_t = \mathbf{e}_m].$$

- The distribution depends on t only through the scalar

$$\lambda_t = 1 - \bar{\alpha}_t \in [0, 1].$$

- We can therefore reparameterize the corruption level by

$$t \quad \Rightarrow \quad \lambda \in [0, 1].$$

- We directly define a family of corrupted distributions indexed by a continuous mask rate λ:

$$q(\mathbf{x}_\lambda|\mathbf{x}_0) = (1 - \lambda)[\mathbf{x}_\lambda = \mathbf{x}_0] + \lambda[\mathbf{x}_\lambda = \mathbf{e}_m].$$

Sahoo S. et al. Simple and effective masked diffusion language models, 2024

Discrete ELBO Revisited

Recall the per-step ELBO term for absorbing diffusion:

$$\mathcal{L}_t = \mathbb{E}_{q(\mathbf{X}_t|\mathbf{X}_0)} \sum_{i=1}^{m} \rho_t [\mathbf{x}_t^i = \mathbf{e}_m] \left[-\log p_\theta(\mathbf{x}_0^i | \mathbf{X}_t) \right] + \text{const.}$$

Discrete ELBO Revisited

Recall the per-step ELBO term for absorbing diffusion:

$$\mathcal{L}_t = \mathbb{E}_{q(\mathbf{X}_t|\mathbf{X}_0)} \sum_{i=1}^{m} \rho_t [\mathbf{x}_t^i = \mathbf{e}_m] \left[-\log p_{\boldsymbol{\theta}}(\mathbf{x}_0^i|\mathbf{X}_t) \right] + \text{const.}$$

Replacing the discrete index t with the continuous mask rate λ, the training objective becomes

$$\mathcal{L} = \int_0^1 w(\lambda) \mathbb{E}_{q_\lambda(\mathbf{X}_\lambda|\mathbf{X}_0)} \sum_{i=1}^{m} [\mathbf{x}_\lambda^i = \mathbf{e}_m] \left[-\log p_{\boldsymbol{\theta}}(\mathbf{x}_0^i|\mathbf{X}_\lambda) \right] d\lambda.$$

Discrete ELBO Revisited

Recall the per-step ELBO term for absorbing diffusion:

$$\mathcal{L}_t = \mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} \sum_{i=1}^{m} \rho_t [\mathbf{x}_t^i = \mathbf{e}_m] \left[-\log p_{\boldsymbol{\theta}}(\mathbf{x}_0^i | \mathbf{X}_t) \right] + \text{const.}$$

Replacing the discrete index t with the continuous mask rate λ, the training objective becomes

$$\mathcal{L} = \int_0^1 w(\lambda) \mathbb{E}_{q_\lambda(\mathbf{x}_\lambda|\mathbf{x}_0)} \sum_{i=1}^{m} [\mathbf{x}_\lambda^i = \mathbf{e}_m] \left[-\log p_{\boldsymbol{\theta}}(\mathbf{x}_0^i | \mathbf{X}_\lambda) \right] d\lambda.$$

Interpretation

Training corresponds to optimizing a **continuous mixture of masked language modeling objectives** with different mask rates.

Algorithm: Masked Diffusion Language Model (MDLM)

Training

1. Sample $\mathbf{X}_0 \sim p_{\text{data}}(\mathbf{X})$ and $\lambda \sim w(\lambda)$.
2. Corrupt the sequence by independent masking:

$$\mathbf{x}_\lambda^i = \begin{cases} \mathbf{e}_m, & \text{with prob. } \lambda, \\ \mathbf{x}_0^i, & \text{with prob. } 1-\lambda, \end{cases} \qquad i = 1, \dots, m.$$

3. Predict token distributions in parallel:

$$p_\theta(\mathbf{X}_0 | \mathbf{X}_\lambda) = \prod_{i=1}^{m} p_\theta(\mathbf{x}_0^i | \mathbf{X}_\lambda).$$

4. Compute the masked-CE loss:

$$\mathcal{L}(\theta) = \sum_{i=1}^{m} [\mathbf{x}_\lambda^i = \mathbf{e}_m] \left[-\log p_{\boldsymbol{\theta}}(\mathbf{x}_0^i | \mathbf{X}_\lambda) \right].$$

Sahoo S. et al. Simple and effective masked diffusion language models, 2024

Algorithm: Masked Diffusion Language Model (MDLM)

Sampling

1. Initialize $\mathbf{X} \leftarrow \mathbf{e}_m \mathbf{1}^\top$ (fully masked).
2. For a decreasing schedule $\lambda_1 > \lambda_2 > \cdots > \lambda_L$:
   2.1 Predict $p_\theta(\mathbf{x}^i | \mathbf{X})$ for all positions.
   2.2 Unmask a subset of positions to reach the next mask rate $\lambda_{\ell+1}$ (e.g., sample tokens for newly-unmasked positions).
3. Return the final sequence $\mathbf{X}$ (fully unmasked).

Outline

Course Overview: Problem Statement

Goal

Learn a generative model $p_{\boldsymbol{\theta}}(\mathbf{x})$ that matches the data distribution $p_{\text{data}}(\mathbf{x})$.

Three similar lenses

- **Divergence minimization:**

$$\min_{\boldsymbol{\theta}} D\left(p_{\text{data}} || p_{\boldsymbol{\theta}}\right) \qquad \text{(KL, JS, Wasserstein, etc.)}$$

- **Likelihood-based modeling:** maximize $\log p_{\boldsymbol{\theta}}(\mathbf{x})$ (NF, VAE, diffusion-as-VAE).
- **Score-based modeling:** learn $\nabla_{\mathbf{x}} \log p(\mathbf{x})$ (DDPM / NCSN / SDE).

What We Covered: Part 1

Likelihood-based

- **Autoregressive (L1)**:

$$p_{\boldsymbol{\theta}}(\mathbf{x}) = \prod_{i=1}^{n} p_{\boldsymbol{\theta}}(x_i|\mathbf{x}_{1:i})$$

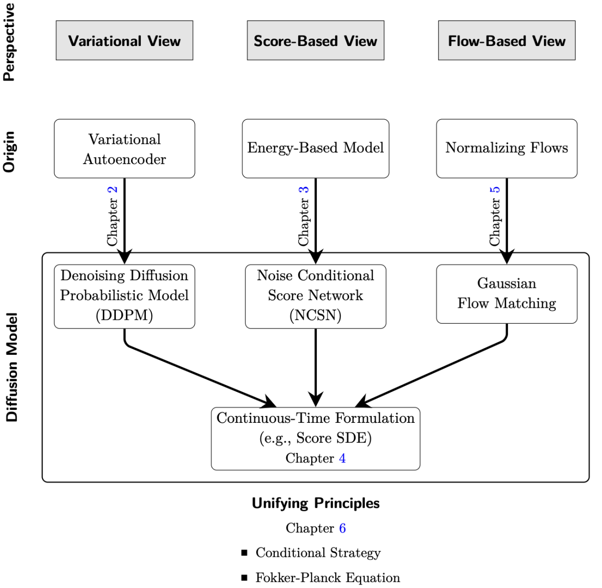
- **Normalizing Flows (L2, L10)**:

$$\mathbf{x} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{z}), \quad \log p_{\boldsymbol{\theta}}(\mathbf{x}) = \log p(\mathbf{z}) + \log \left| \det \frac{\partial \mathbf{z}}{\partial \mathbf{x}} \right|$$

- **VAE (L3–L4)**:

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}) \geq \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})] - \mathrm{KL}(q_{\phi}(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))$$

Generative Models Timeline



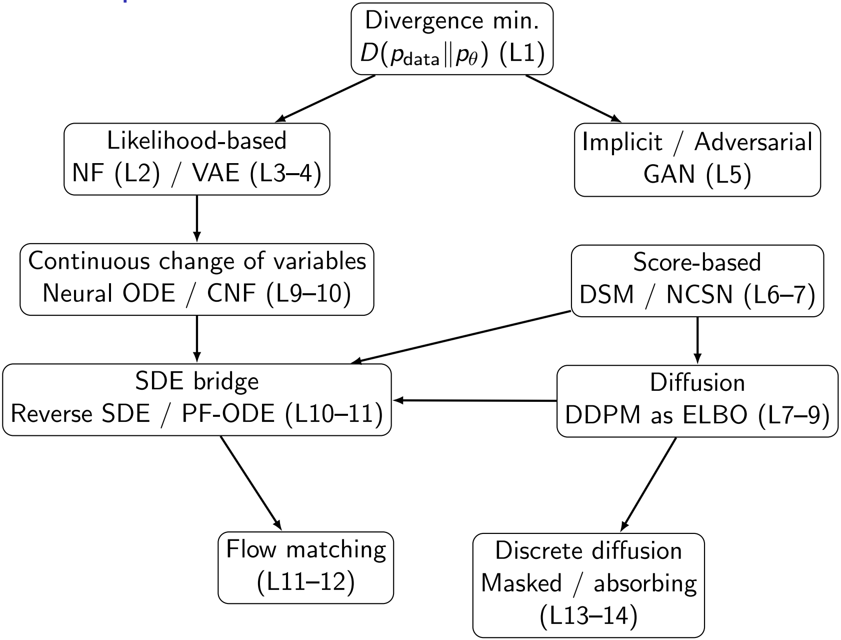
Lai C. H. et al. The principles of diffusion models, 2025.

What We Covered: Part 2

Implicit / Score-based

- **GAN (L5)**: implicit $p_{\boldsymbol{\theta}}$, adversarial learning (close to JSD)
- **Score matching (L6)**: learn $\mathbf{s}_\theta(\mathbf{x}) \approx \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})$
- **Diffusion / DDPM (L7–L9)**: forward noising $q(\mathbf{x}_t|\mathbf{x}_{t-1})$ + reverse denoising
- **SDE / Flow matching (L10–L12)**: reverse SDE $\leftrightarrow$ probability flow ODE, vector field (Neural ODE) / flow matching
- **Discrete diffusion (L13–L14)**: Markov chain on categorical states

Mental Map


Divergence min.
$D(p_{\text{data}} \| p_\theta)$ (L1)
Likelihood-based
NF (L2) / VAE (L3–4)
Implicit / Adversarial
GAN (L5)
Continuous change of variables
Neural ODE / CNF (L9–10)
Score-based
DSM / NCSN (L6–7)
SDE bridge
Reverse SDE / PF-ODE (L10–11)
Diffusion
DDPM as ELBO (L7–9)
Flow matching
(L11–12)
Discrete diffusion
Masked / absorbing
(L13–14)

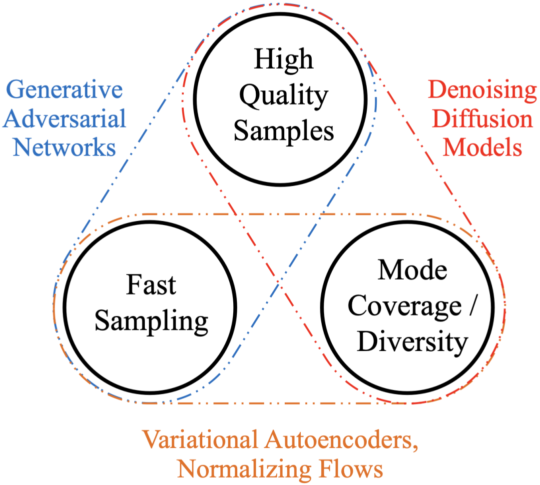
Comparison Cheat-Sheet: Part 1

Family	**Likelihood**	**Training**
AR	✓	stable CE
NF	✓	tricky architecture
VAE	lower bound	stable ELBO
GAN	$\times$	unstable/minimax
Diffusion	bound / est.	stable
FM / ODE	est. / bound	stable
Discr. diff.	bound / CE-like	stable

Comparison Cheat-Sheet: Part 2

Family	**Sampling**	**Best for**
AR	slow (sequential)	text / discrete
NF	fast (1 step)	exact density, OOD
VAE	fast (1 step)	latent modelling
GAN	fast (1 step)	sharp images
Diffusion	slow (many steps)	high fidelity + diversity
FM / ODE	medium–fast	fewer steps + theory
Discr. diff.	iterative	sequences + bidirectional gen

Generative Learning Trilemma



Xiao Z., Kreis K., Vahdat A. Tackling the generative learning trilemma with denoising diffusion GANs, 2021

Generative Learning Trilemma

Rule of Thumb

- **Likelihood** & **Coverage** $\Rightarrow$ **AR / NF** exact density, no mode dropping, *slow sampling*
- **Likelihood** & **Fast Sampling** $\Rightarrow$ **VAE** tractable bound, fast, *blurry samples*
- **Sample Quality** & **Fast Sampling** $\Rightarrow$ **GAN** sharp samples, *no likelihood, mode collapse*
- **Quality** & **Coverage** $\Rightarrow$ **Diffusion** stable training, high fidelity, *slow sampling*
- **Quality** & **Faster Sampling** $\Rightarrow$ **FM / ODE** fewer steps, continuous flows, *approx. likelihood*
- **Discrete Structure** & **Coverage** $\Rightarrow$ **Discrete Diffusion** stable CE training, parallel denoising, *iterative decoding*

Summary

- For sequences, the forward process of discrete diffusionfactorize over positions, but reverse process (the model $p_{\boldsymbol{\theta}}$) conditions on the whole context.
- In the absorbing case, tokens are either unchanged or masked; so only masked tokens contribute to the ELBO loss.
- The discrete ELBO reduces to a MLM objective.
- Reparameterizing time by the mask rate $\lambda \in [0, 1]$ yields a continuous mixture of MLM losses.
- MDLM sampling performs iterative parallel refinement from fully masked to fully unmasked sequences.
- No generative model is strictly better than all others: different methods occupy different corners of the generative learning trilemma and come with unavoidable disadvantages.