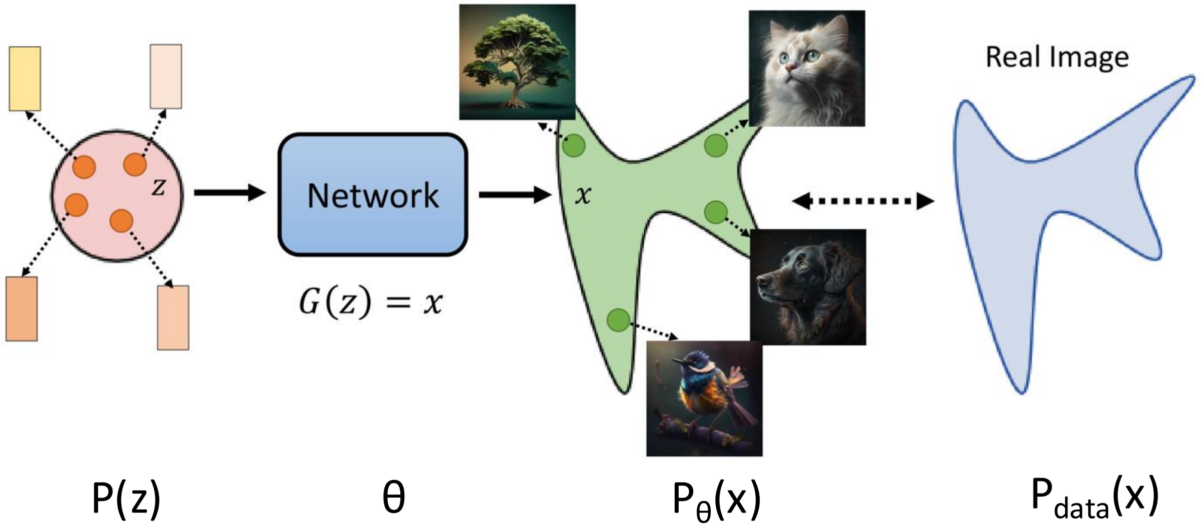
Large Language Diffusion Models

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, Chongxuan Li

Presenter: Ye YUAN

Contents

Preliminary – Likelihood-based Generative Models



Sample $\{x^1, x^2, \ldots, x^m\}$ from $P_{data}(x)$

$$\theta^* = arg\max_\theta \prod_{i=1}^{m} P_\theta(x^i) \; = arg\max_\theta log \prod_{i=1}^{m} P_\theta(x^i)$$

$$= arg\max_\theta \sum_{i=1}^{m} logP_\theta(x^i) \; \approx arg\max_\theta E_{x\sim P_{data}}[logP_\theta(x)]$$

(not related to θ)

$$= arg\max_\theta \int_x P_{data}(x)logP_\theta(x)dx \; - \int_x P_{data}(x)logP_{data}(x)dx$$

Difference between P_{data} and P_θ

$$= arg\max_\theta \int_x P_{data}(x)log\frac{P_\theta(x)}{P_{data}(x)}dx \; = arg\min_\theta KL(P_{data}||P_\theta)$$

Maximum Likelihood = Minimize KL Divergence

- $P_{data}(x)$: probability distribution of the data
- $P_\theta(x)$: approximate probability distribution of the data
- $P(z)$: probability distribution of the latent variable, usually a Gaussian distribution

[1] Luo, C. Understanding diffusion models: A unified perspective, 2022.

Preliminary – Evidence Lower Bound (ELBO)

- In practice, we usually maximize the log likelihood.
- Sometimes, the log likelihood is also called evidence.
- We can maximize the ELBO instead.

$$
\begin{aligned}
\log p(x) &= \log p(x) \int q_\phi(z|x)dz \\
&= \int q_\phi(z|x)(\log p(x))dz \\
&= \mathbb{E}_{q_\phi(z|x)}[\log p(x)] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p(x,z)}{p(z|x)}\right] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p(x,z)q_\phi(z|x)}{p(z|x)q_\phi(z|x)}\right] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p(x,z)}{q_\phi(z|x)}\right] + \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{q_\phi(z|x)}{p(z|x)}\right] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p(x,z)}{q_\phi(z|x)}\right] + D_{KL}(q_\phi(z|x)||p(z|x)) \\
&\geq \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p(x,z)}{q_\phi(z|x)}\right].
\end{aligned}
$$

$$p(x) = \frac{p(x,z)}{p(z|x)}$$

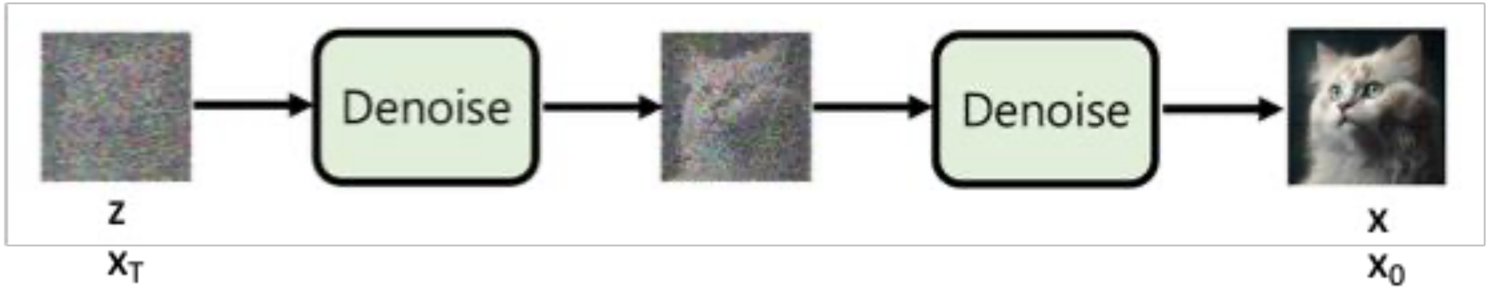
Evidence Lower Bound (ELBO)

[1] Luo, C. Understanding diffusion models: A unified perspective, 2022.

Preliminary – Denoising Diffusion Models

- Diffusion Process (Forward Process)

- Denoising Process (Backward Process)



Preliminary – ELBO of Diffusion Models

$\mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_{\theta}(\boldsymbol{x}_0|\boldsymbol{x}_1)\right]$: a reconstruction term, predicting the log probability of the original data sample given the first-step latent.

$D_{KL}(q(\boldsymbol{x}_T|\boldsymbol{x}_0)||p(\boldsymbol{x}_T))$: represents how close the distribution of the final noisified input is to the standard Gaussian prior.

$\sum_{t=2}^{T}\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[D_{KL}(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)||p_{\theta}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t))\right]$: a denoising matching term. The $q(x_{t-1}|x_t, x_0)$ defines how to denoise a noisy image x_t with access to what the final, completely denoised image x_0 should be.

$$\log p(\boldsymbol{x}) = \log \int p(\boldsymbol{x}_{0:T})d\boldsymbol{x}_{1:T}$$

$$\geq \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_{0:T})}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\right]$$

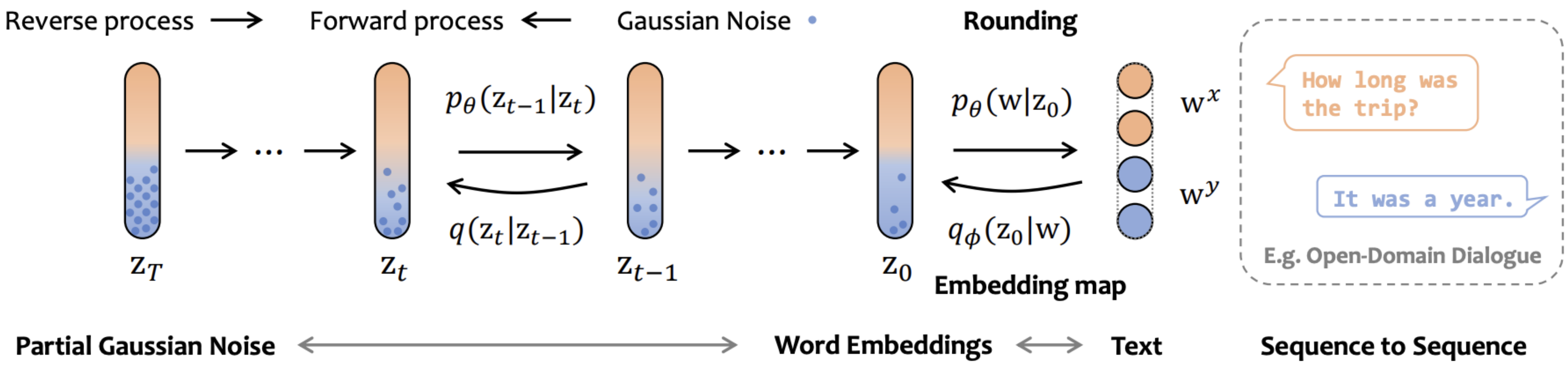
$$= \mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_{\theta}(\boldsymbol{x}_0|\boldsymbol{x}_1)\right] - D_{KL}(q(\boldsymbol{x}_T|\boldsymbol{x}_0)||p(\boldsymbol{x}_T))$$

$$- \sum_{t=2}^{T}\mathbb{E}_{\boxed{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}}\left[D_{KL}(\boxed{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}||p_{\theta}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t))\right].$$

[1] Luo, C. Understanding diffusion models: A unified perspective, 2022.
[2] Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. In Advances in Neural Information Processing Systems (2020).

Diffusion for Language Modeling (Continuous)

- Convert to Continuous State Space:



[3] S. Gong, M. Li, J. Feng, Z. Wu, and L. Kong, DiffuSeq: Sequence to Sequence Text Generation with Diffusion Models. International Conference on Learning Representations (2023).

Diffusion for Language Modeling (Discrete)

- Reformulate Two Key Equations in Discrete State Space:

  Suppose we have a vocabulary table of size $K = 4$: “apple”, “banana”, “carrot”, “[MASK]”.

  Define a transition matrix $Q_t \in \mathbb{R}^{K \times K}$ at time t that govern the corruption process:

$$Q_t = \begin{bmatrix} 0.1 & 0.1 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 & 0.7 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix}$$

  This means any normal token (e.g., “banana”) has a 70% chance of turning into [MASK] at time t. Once a token becomes [MASK], it stays that way.

[4] J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and van, Structured Denoising Diffusion Models in Discrete State-Spaces. In Advances in Neural Information Processing Systems (2021).

Diffusion for Language Modeling (Discrete)

"apple", "banana", "carrot", "[MASK]".

$$Q_t = \begin{bmatrix} 0.1 & 0.1 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.1 & 0.7 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix}$$

- Forward Process:

$$q(x_t \mid x_{t-1}) = \mathrm{Cat}(x_t;\, p = x_{t-1}Q_t)$$

x_{t-1} is a one-hot row vector. For "banana", $x_{t-1} = [0, 1, 0, 0]$.

$x_{t-1}Q_t = [0.1, 0.1, 0.1, 0.7]$: give the probability of transitioning to other tokens.

- $q(x_t \mid x_0) = \mathrm{Cat}\left(\boldsymbol{x}_t; \boldsymbol{p} = \boldsymbol{x}_0\overline{\boldsymbol{Q}}_t\right), \quad \text{with} \quad \overline{\boldsymbol{Q}}_t = \boldsymbol{Q}_1\boldsymbol{Q}_2\ldots\boldsymbol{Q}_t$

- $q(x_{t-1} \mid x_t, x_0) = \dfrac{q(x_t \mid x_{t-1}, x_0)\, q(x_{t-1} \mid x_0)}{q(x_t \mid x_0)} = \mathrm{Cat}\left(\boldsymbol{x}_{t-1}; \boldsymbol{p} = \dfrac{\boldsymbol{x}_t\boldsymbol{Q}_t^\top \odot \boldsymbol{x}_0\overline{\boldsymbol{Q}}_{t-1}}{\boldsymbol{x}_0\overline{\boldsymbol{Q}}_t\boldsymbol{x}_t^\top}\right)$

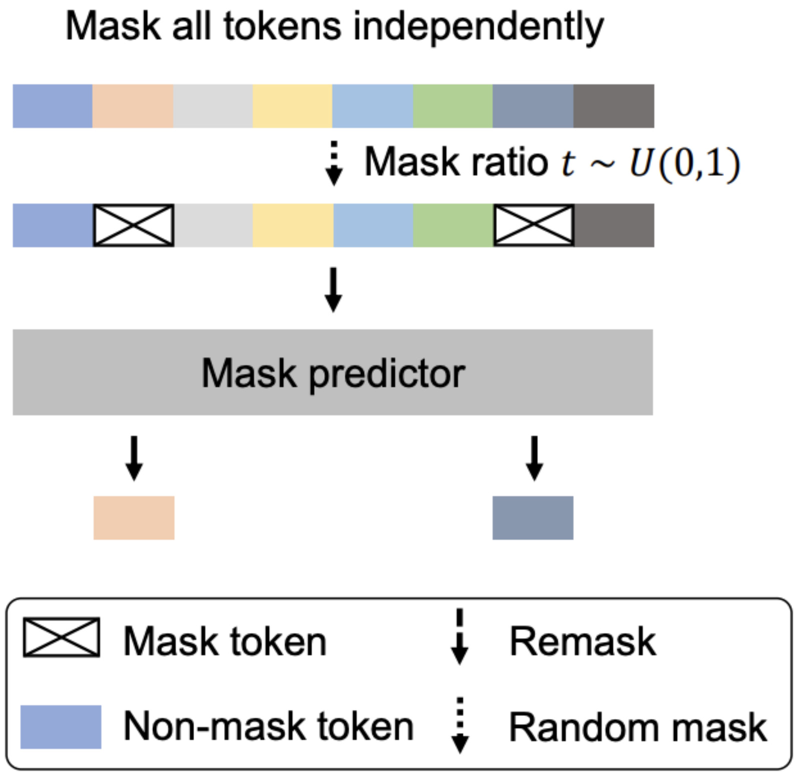
how likely each candidate x_{t-1} would result in the current x_t

how likely x_t was generated from the original input x_0

how likely each x_{t-1} was generated from the original input x_0

Large Language Diffusion Models - Pretraining

- This paper introduces large language diffusion model (LLaDA), with discrete states formulation. Each token can only be corrupted to [MASK] token in this paper.

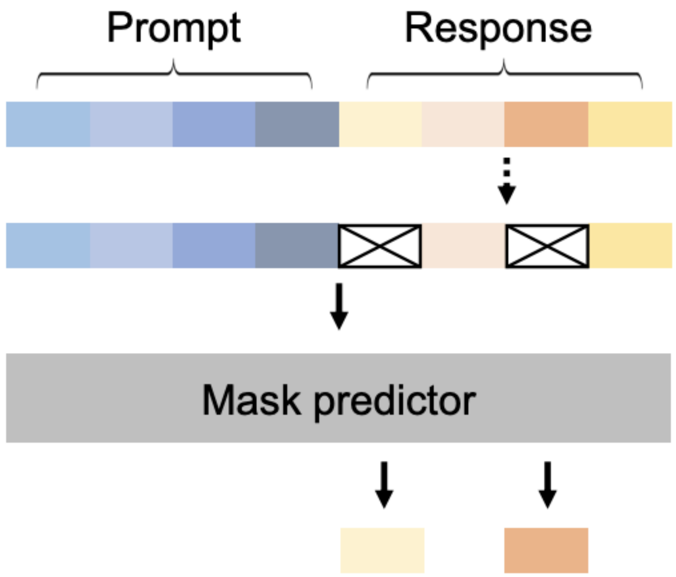


$$x_0 \sim p_{\text{data}}, t \sim \mathrm{U}(0,1]$$
$$x_t \sim q_{t|0}(x_t|x_0)$$

This is like the BERT pretraining phase but with random mask ratio between 0 to 1.

Large Language Diffusion Models – SFT

- To enhance the instruction following capability of LLaDA, the authors perform supervised fine-tuning, similar to other autoregressive LLMs like Llama.

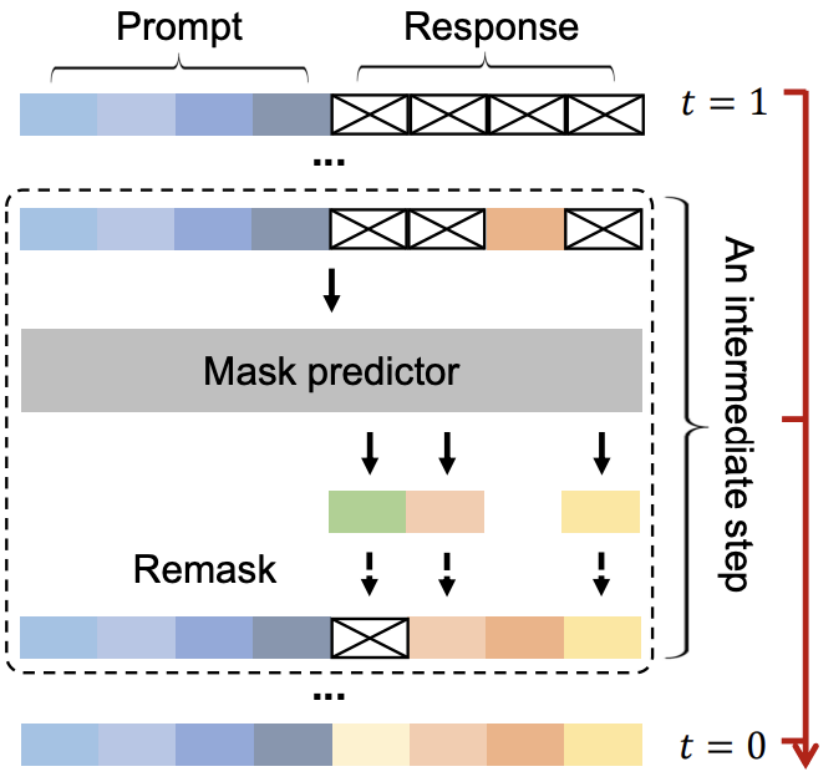


$$p_0, r_0 \sim p_{\text{data}}, t \sim \mathrm{U}(0, 1]$$
$$r_t \sim q_{t|0}(r_t | r_0)$$

p_0 is a prompt, r_0 is a response.
Only apply masking to response.

Large Language Diffusion Models - Sampling

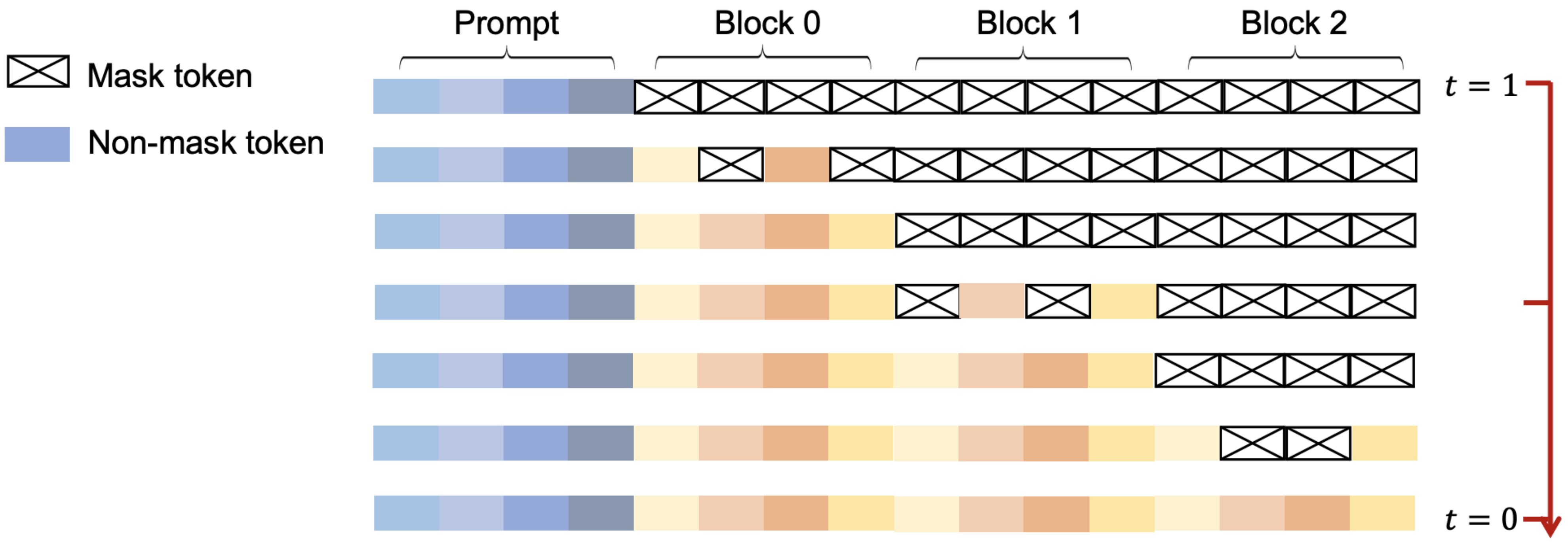
- LLaDA predicts all [MASK] tokens at once, then remasks some tokens and predicts them again. By doing this, the model can iteratively generate a response.



- Initialize the response as a sequence of [MASK] tokens with length s. The total steps for generation is T. The time steps could be $[0, {}^{1}/_{T}, \ldots, 1]$, and we remask $s \times {}^{i}/_{T}$ tokens at discrete time step i.

- Three different remask strategies:
  - **Random**: randomly remask tokens.
  - **Low-Confidence**: remask lowest confidence tokens.
  - **Semi-Autoregressive**: divide the sequence into several blocks and generate them from left to right.

Semi-Autoregressive Remasking



Discussion and Conclusion

- Advantages of LLaDA:
  - Faster sampling than autoregressive model.
  - Better reversal reasoning ability.
- Disadvantages of LLaDA:
  - Harder to calculate the probability of a generated sequence. (Need Monte-Carlo Estimations)
  - Generated sequence length and total generation steps are hyperparameters.
  - Still lower performance on general tasks than autoregressive LLMs.

Table 3. **Comparison in the Poem Completion Task.**

	Forward	Reversal
GPT-4o (2024-08-06)	**82.7**	34.3
Qwen2.5 7B Instruct	75.9	38.0
LLaDA 8B Instruct	48.8	**42.4**

Thanks for your attention!

References

[1] Luo, C. Understanding diffusion models: A unified perspective, 2022.
[2] Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. In Advances in Neural Information Processing Systems (2020).
[3] S. Gong, M. Li, J. Feng, Z. Wu, and L. Kong, DiffuSeq: Sequence to Sequence Text Generation with Diffusion Models. International Conference on Learning Representations (2023).
[4] J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and van, Structured Denoising Diffusion Models in Discrete State-Spaces. In Advances in Neural Information Processing Systems (2021).
[5] S. Nie et al., “Large Language Diffusion Models,” 2025.