Paper summary

AI VISION Lab

1. 공부한 논문의 제목, 게재된 학회 혹은 저널 등 논문 기본 정보를 적으세요.
   A. **이름: Score-Based Generative Modeling Through Stochastic Differential Equations**
   B. **저널: ICML**
   C. **도메인: Score-based Generative model**
   D. **출판연도: 2021**
   E. **저자: Yang Song, Sohl-Dickstein, P. Kingma, Kumar, Ermon, Poole**

2. 논문에서 제안한 알고리즘 및 프레임워크에 대해 본인이 이해한대로 다이어그램을 그려보세요. **논문 Figure를 그대로 따라 그리면 안됩니다.**
   A. [선행연구] SMLD(Score-Matching Langevin Dynamics)의 경우, Score Matching을 기반으로 하는 Loss를 사용. 또한, 학습한 Model을 기반으로, Annealed Langevin Dynamics를 통한 Sampling으로 구성.

   B. [선행연구] DDPM의 경우, Score-matching을 사용한다고 명시적으로 설명하지는 않고 있으나, Log-likelihood를 구하는 Diffusion model의 특성상, Score-Matching의 형태와 비슷한 구조를 가지고 있음.

   C. 이 논문에서는 두 선행연구를 Score-Matching Based Generative model이라 이야기하되, Noise를 추가/제거하던 과정이 이산적이었던 것을 (1, 2, 3, …, N) SDE(Stochastic Differential Equations)기반의 형태로, 연속적으로 표현하고자 함. 뿐만 아니라, 두 모델에서 동시에 사용하던 Score Matching을 기준으로, 두 가지 형태의 Noise 주입 형태(VE/VP)를 제안함.

   D. $p(x_t \mid x_{t-1})$의 형태처럼, t-1 - >t 형태의 Forward Process를 Discrete model의 특징이라고 이야기 했을 때, 우리는 이것의 분포를 정확하게 표현할 수 있음(Gaussian Distribution). DDPM논문에서는 해당 Distribution을 Reparameterization하여, Diffusion항과, x_{t-1}을 scaling하는 항(고정)으로 구성했다면 본 논문에서는 이를 Drift항과 Diffusion항으로 구성함(Ito Process).

   E. $t-1 \rightarrow t$의 형태를, $t-\delta x \rightarrow t$ 로 근사함으로써, 연속적인 분포를 따르도록

했고, 결과적으로 SDE를 해결하고자 적용되었던 다양한 solver(e.g., Euler-Maruyama)를 사용할 수 있는 Framework를 사용할 수 있게 됨.

F. Forward process뿐만 아니라, 선행연구 (1982, Anderson)에서 제시된, Reverse Diffusion 수식을 적용할 수 있게 됨으로써, 기존의 Reverse Process 역시 SDE형태로 해결할 수 있게 됨.

G. 또한, 선행연구 SMLD, DDPM의 서로 다른 Noise 주입과정에 대해서, 다른 형태의 SDE 수식을 보임. 이는, VE, VP 뿐만 아니라, 추가적인 SDE가 Diffusion 과정에 적용이 가능함을 의미함.

H. 심지어 Sampling과정에서 SMLD: Annealed Langevin Dynamics, DDPM: Ancestral Sampling의 형태로 고정된 구조가 아닌, PC sampling 형태를 제시함으로써, 원하는 SDE-Solver로 Sampling 함수를 변경할 수 있는 확장성 있는 구조를 제시함.

I. 결과적으로는, 확장성 뿐만 아니라, 성능의 향상까지 기여했다는 점에서 이 논문은 큰 의미가 있다고 할 수 있음.

3. 본인이 생각하는 이 논문의 장점이 무엇이라고 생각하나요? **논문 Contribution bullet을 그대로 따라 적으면 안됩니다.**
   A. **확장성. SDE를 결합함으로써, 기존에 존재하던 하나의 길이 아닌, SDE의 어떤 함수라도 적용할 수 있는 Framework를 제시했다는 점이 이 논문의 최대 장점이라 할 수 있음.**

4. 이 논문을 읽으면서 느낀 점, 혹은 배운 점이 있으면 적어보세요.
   A. 현 시점의 논문을 이해하는데 큰 도움이 될 것이라 판단됨.

5. 이 논문의 한계점이 있다면 무엇이라고 생각하나요?
   A. 확장성은 제시했으나, 다양한 SDE를 활용해 실험하는 과정이 상대적으로 부족함.

6. 본인의 연구에 접목시켜볼 점이 있을지 생각하고 적어보세요.
   A. Diffusion 선행연구

7. 본 Summary를 작성하는 과정에서 생성형AI를 사용했나요?
    A. 아니요

날짜: 2025-07-03

이름: 신준원

07/03 Score-based Generative modeling Through Stochastic Diffrential Equations

Introduction

X (input) + 순차적인 noise 주입 과정.] → DDPM, SMLD (=NCSNs)

(추정)

주입된 noise 를 예측/제거 하는 과정]

= 우수한 성능을 보인 Generative models

① SMLD (=NCSNs)

→ 각 레이어에 대한 Score function (= Score 추정)

⇒ each noise Scale ($x_0 + \varepsilon_1 = x_1$ → x_0 Score, x_1 Score, x_2 Score, ..., x_n Score)

→ input 1개 다양한 step / 모든 step Score estimation (with Score function)

→ Langevin dynamics 로 Sampling 진행

$$x_{k+1} = x_k + \frac{\varepsilon}{2} \underbrace{\nabla_x \log p(x_k)}_{\text{Score function}} + \sqrt{\varepsilon} \cdot z_k \qquad z_k \sim N(0, I)$$

<Langevin Dynamics ⇒ 즉, Score function 기반의 경우 → 곧바로 x_{k+1} Sampling 가능.

② DDPM

→ foward step 과정의 noise 추정

⇒ 이를 역방향으로 reverse ($p_\theta(x_{t-1} \mid x_t)$) 에 대한

Closed-form 제시 + 학습.

이때 DDPM ⇒ Score function 을 계산. (implicit)

↳ log-likelihood 형태 계산. = $\nabla \log p_\theta(x)$!.

결론

DDPM ⇒ Implicit 한 Score-function 이용 } → "Score-based generative models"

SMLD ⇒ explicit 한 Score-function 이용

본 논문에서는 SDE 를 사용한 SBM (Score-based-model) 을 제안함

통합된 구조

~~본 논문에서는 두 SBM (DDPM, SMLD)를 통합하고 SDE 사용~~

$\boxed{\text{SDE 기반의 SBM}}$ → 최종목표

$\boxed{\text{with SDE,}}$ 학습 X, data X → 연속적으로

$\boxed{\text{과거 모델 (DDPM, SMLD)}}$

- 불연속적인 noise 주입 ⇒ (i = integer, x_i, x_{i+1} ...) → 연속적으로

$\boxed{\text{새로운 제안 with SDE}}$

- 연속적인 구조 제안.
  - → Sample Generation 과정을 smooth 하게
  - → time(t) 에 의존 하는 구조.
    - ⇒ reverse process 에 필요한 정보 from(forward SDE 의 marginal 분포)
  - ⇒ Sampling 에 다양한 SDE 기법 적용가능

④ Contributions

① flexible sampling ~~with~~ & loglikelihood Computation
② Controllable Generation
③ Unified - framework

Background

① SMLD (=NCSN)

- perturbation kernel (= forward kernel)

$\boxed{p_\sigma(\tilde{x}|x) = \mathcal{N}(\tilde{x}; x, \sigma^2 I)}$, $p_\sigma(\tilde{x}) = \int p_{data}(x) \cdot p_\sigma(\tilde{x}|x) dx$

$\rightarrow \tilde{x} = x + \sigma I.$ $(\sigma_{min} = \sigma_1 < \sigma_2 < \cdots < \sigma_N = \sigma_{max})$

$p_{\sigma_{min}}(x) \approx p_{data}(x)$ $\quad p_{\sigma_{max}}(x) \approx \mathcal{N}(x; 0, \sigma_{max}^2 I)$

- optimizer $\rightarrow \sigma_i^2 \propto \dfrac{1}{E[\|\nabla_x \log p_{\sigma_i}(\tilde{x}|x)\|_2^2]}$

$$\theta^* = \arg\min_\theta \sum_{i=1}^{N} \sigma_i^2 E_{p_{data}(x)} E_{p_{\sigma_i}(\tilde{x}|x)} \left[\| S_\theta(\tilde{x}, \sigma_i) - \nabla_{\tilde{x}} \log p_{\sigma_i}(\tilde{x}|x) \|_2^2 \right]$$

- Sampling (Langevin Dynamics)

$$x_i^m = x_i^{m-1} + \varepsilon_i S_{\theta^*}(x_i^{m-1}, \sigma_i) + \sqrt{2\varepsilon_i}\, z_i^m \quad m = 1, 2, \cdots, M$$

$z_i \sim \mathcal{N}(0, I)$

$\rightarrow i = N$ 일때 , M번

$i = N-1$ 일때 , M번

⋮

$i = 1$ 일때 , M번

$N \times M$번 반복 $\Rightarrow x_0^0 \sim \mathcal{N}(x|0, \sigma_{max}^2 I)$

$x_i^0 = x_{i+1}^M$

ⓥ DDPM

- forward process

$$\boxed{P(x_i | x_{i-1}) = \mathcal{N}(x_i ; \sqrt{1-\beta_i}\, x_{i-1}, \beta_i I)}$$

$$P_{\alpha_i}(x_i | x_0) = \mathcal{N}(x_i ; \sqrt{\alpha_i}\, x_0, (1-\alpha_i) I), \quad \alpha_i = \Pi_{j=1}^{i} (1-\beta_j)$$

$$P_{\alpha_i} = \int P_{data}(x)\, P_{\alpha_i}(\tilde{x}|x)\, dx$$

$x_T \sim \mathcal{N}(0, I)$

- reverse process

$$P_\theta(x_{i-1} | x_i) = \mathcal{N}\left(x_{i-1} ; \frac{1}{\sqrt{1-\beta_i}}(x_i + \beta_i s_\theta(x_i, i)), \beta_i I\right)$$

- optimizer (L_{simple}) $\rightarrow (1-\alpha_i) \propto \frac{1}{E[\| \nabla_x \log P_{\alpha_i}(\tilde{x}|x) \|_2^2]}$

$$\theta^* = \underset{\theta}{\arg\min} \sum_{i=1}^{N} (1-\alpha_i) E_{P_{data}(x)} E_{P_{\alpha_i}(\tilde{x}|x)} \left[\| s_\theta(\tilde{x}, i) - \nabla_{\tilde{x}} \log P_{\alpha_i}(\tilde{x}|x) \|_2^2 \right]$$

동일 ☆

- Sampling $\approx x_T$. (DDPM의 경우 x_0부터, DDIM은 $\rightarrow x_T$) 곧로 closed form 형태 계산가능.

input : $x_N \sim \mathcal{N}(0, I)$.

$$x_{i-1} = \frac{1}{\sqrt{1-\beta_i}}(x_i + \beta_i s_{\theta^*}(x_i, i)) + \sqrt{\beta_i}\, z_i \quad i = N, N-1, \dots 1.$$

trained model

x_i : input, x_{i-1} : output $\rightarrow$ $\boxed{\text{ancestral sampling}}$

SBM with SDEs

목표 : 연속적인 구조. ($x_1, x_2, \dots$ 가 아닌 $x(0)$)

= Brownian motion

→ 주어진 dataset
→ p_0 = data distribution
p_T = prior distribution

→ Itô SDE
$$dx = \underbrace{f(x,t)}_{\text{drift}}dt + \underbrace{g(t)}_{\text{diffusion}}dw$$
→ Wiener process
drift → vector-valued
diffusion → Scalar function

$\Rightarrow x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\,\varepsilon$

Drift function / diffusion function

Q) SDE 구조로 표현할때 얻을 수 있는 장점
① 연속적인 forward process
② Stochastic 한 random function 의 움직임을 deterministic하게 표현가능
⇒ State(i), time(t) 가 Lipschitz 조건을 만족하는 경우,

($\|f(x,t) - f(y,t)\| \le \underbrace{L}_{\text{상수}} \|x-y\|$ 모든 x, y, t 에 대해)

즉, $L \ge \dfrac{\|f(x,t) - f(y,t)\|}{\|x-y\|}$

ex) $x_t = \sqrt{1-\beta_t}\, x_{t-1} + \sqrt{\beta_t}\,\varepsilon$.
$f(x,t) = \sqrt{1-\beta_t}\, x_{t-1}$, $g(x,t) = \sqrt{\beta_t}\,\varepsilon$.
$x_t = f(x,t) + g(t)$
→ 변수 a, b에 대해서 Lipschitz 조건 만족확인

$$\|f(a) - f(b)\| = \|\sqrt{1-\beta_t}\, a - \sqrt{1-\beta_t}\, b\|$$
$$= \sqrt{1-\beta_t}\, \|a-b\|$$

$\sqrt{1-\beta_t}$ 는 유한한 값. (β_t 고정 fix)

Generate samples by reversing SDE

denote as..

$P_t(x) = X(t)$

$P_{st}(X(t)|X(s)) = \text{kernel}(S \to t)$

$S_\theta(x, t) = \nabla_x \log P_t(x)$

$X_T \to X_0$ (sample).

$$\boxed{dx = [f(x,t) - g(t)^2 \underbrace{\nabla_x \log P_t(x)}_{\Rightarrow S_\theta^*(x,t)}]\, dt + g(t) d\bar{w}}$$

(by. Anderson 1982)

dt → negative timestep

$d\bar{w}$ → backward wiener process

dx → Reverse Diffusion SDE

Estimate Scores for the SDE

일반적으로, Score of model ($= \nabla_x \log P_t(x)$) 는 계산 불가능

따라서. $\theta^* = \arg\min_\theta E_t \{ \lambda(t) E_{X(0)} E_{X(t)|X(0)} [\| S(\theta)(X(t), t) - \nabla_{X(t)} \log P_{0t}(X(t)|X(0)) \|_2^2] \}$ 사용

$\lambda(t)$:

positive weight function → σ_i^2 or $(1-\alpha_i)$

Continuous = discrete

→ $\lambda \propto \frac{1}{E[\| \nabla_{X(t)} \log P_{0t}(X(t)|X(0)) \|_2^2]}$

P_{0t} → $\boxed{0 \text{ to } t}$ $\boxed{\text{연속적}}$

일반적 model → 이산적인 값 $\boxed{t \to t+1 \text{ 값}}$

현재 SDE based → $\boxed{0 \to t \text{ 값}}$.

그렇다면 해당 kernel 은 어떻게 구하는가?

→ $f(\cdot, t)$ 가 affine $\{y = ax + b\}$ 형태라면 = Gaussian 분포 (kernel 이)

→ 평균, 분산. - closed form

⊕

혹은, kolmogorov's forward equation 사용

혹은, $P_{0t}(X(t)|X(0))$ sample 해서 SDE 대체

혹은, sliced Score Matching 사용 ⇒ $\nabla_{X(t)} \log P_{0t}(X(t)|X(0))$ 계산 skip. . .

SDE ⇒ 다양한 function의 적용이 가능해짐!

☆ Diffusion Bridge 에도 나오는 내용

$\boxed{\text{VE, VP and BeYond}}$.

① SMLD (VE)

perturbed kernel

$$x_t = x_{t-1} + \sqrt{\sigma_t^2 - \sigma_{t-1}^2}\, z_{t-1}, \quad t = 1, \dots, N.$$

- $\sigma_0 = 0$, $z_{t-1} \sim N(0, I)$
- $N \to \infty$, $\{\sigma_t\}_{t=1}^{N} \to \sigma(t)$, $z(t)$. = 연속
- markov chain → Continuous Stochastic process 변화.

$$dx = \sqrt{\frac{d[\sigma^2(t)]}{dt}}\, dw, \quad \{x(t)\}_{t=0}^{1} \text{ 일때.}$$

$x(t)$는 고정, 변화는 diffusion 만. = $\boxed{\text{VE (variance exploding)}}$

② DDPM (VP)

$$x_t = \sqrt{1-\beta_t} \cdot x_{t-1} + \sqrt{\beta_t} \cdot z_{t-1}, \quad t = 1, \dots, N.$$

- $\beta_t \rightarrow \beta(t)$, $z_{t-1} \rightarrow z(t)$ = "연속"

$i-1$ to i / $i-\Delta t$ to i / i to $i+\Delta t$.

derivation) $x(t+\Delta t) = \sqrt{1-\beta(t+\Delta t)\Delta t}\, x(t) + \sqrt{\beta(t+\Delta t)\Delta t}\, z(t)$

taylor 전개

1차근사 $\approx x(t) - \frac{1}{2}\beta(t+\Delta t)\Delta t\, x(t) + \sqrt{\beta(t+\Delta t)\Delta t}\, z(t)$

0차근사 $\approx x(t) - \frac{1}{2}\beta(t)\Delta t\, x(t) + \sqrt{\beta(t)\Delta t}\, z(t)$

$$\boxed{dx = -\frac{1}{2}\beta(t)\, x\, dt + \sqrt{\beta(t)}\, dw}$$

Δt 작은 경우

fixed 된 variance → $\boxed{\text{VP (variance preserving)}}$

③ Sub-VP SDE (제안하는 새로운 SDE)

$$dx = -\frac{1}{2}\beta(t)\, x \cancel{(t)}\, dt + \sqrt{\beta(t)\left(1 - e^{-2\int_0^t \beta(s)ds}\right)}\, dw$$

- VP 와 동일한 $\beta(t)$ 사용
- $\sqrt{\beta_t}$ $(t=0)$ 를 [illegible] [illegible] 구조 (그래서 sub-VP)

Solve the reverse SDE

~ from p_0

model training 다음엔 → sampling! (x_0)

- Euler - Maruyama
- Stochastic Runge - kutta method
- Ancestral sampling
- etc (다른 SDE 해결방안 전부 적용가능)

Predictor - Corrector

일반적인 SDE
\+
S(θ)의 성질 ⇒ MCMC 적용 편리.

↳ Sampling (from p_t) ⇒ (수많은) SDE 해결방안에서 나온 해를 모을 수 있음.
one step

Predictor : SDE - solvers
Corrector : any score - based MCMC approach.

즉, predictor 가 생성한 sample 기반 distribution

↳ Corrector 가 check

결과 ⇒ PC - sampler 성능↑.

probability flow & ODE

SDE와 같은, 그러나 deterministic한 ODE → 모든 diffusion process에 존재.

$$dx = \left[f(x,t) - \frac{1}{2} g(t)^2 \nabla_x \log p_t(x) \right] dt \rightarrow \boxed{dw\ X}$$

→ probability flow ODE.

→ SDE와 같은 흐름, but deterministic.

ODE가 가지는 장점

- Random성 X.
- → 정확한 likelihood 계산 → $\boxed{\text{결과 동일}}$.
- → Image editing, interpolation, temperature Scaling 사용가능