

Classifier - Free Guidance Diffusion Model 노트

1. 개요

- **목표:** 단일 네트워크로 **조건부**와 **무조건부** score/노이즈 예측을 모두 학습하여, 샘플링 시 조건 반영 강도를 자유롭게 조절
 - **핵심 기법:** 학습 단계에서 조건 드롭 아웃, 샘플링 단계에서 조건/무조건 두 번 호출 후 가중합
-

2. 학습(Training)

2.1 ϵ (노이즈) 예측 기반 (DDPM)

1. **노이즈:** $x_t = \sqrt{\alpha_t}x_0 + \sqrt{1 - \alpha_t}\epsilon$, $\epsilon \sim \mathcal{N}(0, I)$
2. **조건 드롭아웃:**
3. 입력 조건 y 를 확률 p_{drop} 로 \emptyset 로 교체
4. $y_{in} = y$ 시 조건부 $\epsilon_\theta(x_t, t, y)$ 학습
5. $y_{in} = \emptyset$ 시 무조건부 $\epsilon_\theta(x_t, t, \emptyset)$ 학습
6. **손실:** $\mathcal{L} = \mathbb{E}\|\epsilon - \epsilon_\theta(x_t, t, y_{in})\|^2$

2.2 Score 예측 기반 (Score Matching)

1. **노이즈:** (위와 동일)
2. **조건 드롭아웃:** ϵ 예측과 동일하게 적용
3. **Score 네트워크:** $s_\theta(x_t, t, y_{in}) \approx \nabla_{x_t} \log p(x_t | y_{in})$
4. **손실 (DSM):** $\mathcal{L} = \mathbb{E}\|s_\theta(x_t, t, y_{in}) + \epsilon/\sqrt{1 - \alpha_t}\|^2$

하나의 네트워크로 조건부/무조건부 두 모드를 번갈아가며 학습\ Twin - head 구조로 한 번의 포워드에서 두 예측을 동시에 뽑아내는 구현도 가능

3. 샘플링(Sampling) & Classifier-Free Guidance

1. **초기화:** $x_T \sim \mathcal{N}(0, I)$
 2. **각 스텝** $t = T, \dots, 1$ **:
 3. **무조건부 예측**
 - ϵ 기반: $\hat{\epsilon}_{un} = \epsilon_\theta(x_t, t, \emptyset)$
 - Score 기반: $s_{un} = s_\theta(x_t, t, \emptyset)$
 4. **조건부 예측**
 - ϵ 기반: $\hat{\epsilon}_{cond} = \epsilon_\theta(x_t, t, y)$
 - Score 기반: $s_{cond} = s_\theta(x_t, t, y)$
 5. **Guidance 가중합:** $\hat{u} = u_{un} + r(u_{cond} - u_{un})$
 - u 는 ϵ 또는 score
 - r (또는 γ): guidance scale (조건 강화 강도)
 6. **디노이징 업데이트:** DDPM/DDIM/Euler-Maruyama 공식에 \hat{u} 적용
 7. **완료:** x_0 디노이징 후 샘플 반환
-

4. 조건/무조건 점수 분리의 필요성

조건부와 무조건부 점수를 분리하여 학습하는 이유는 다음 두 가지입니다:

1. **Guidance 시 baseline 역할 확보**
2. 샘플링 시 조건부 효과만 추출하려면, 조건부 점수에서 무조건부 점수를 빼야 함
3. 따라서 무조건부 점수가 **baseline** 역할을 하며, 이를 위해 생성 시에도 무조건부 점수를 예측할 수 있어야 함
4. 즉, 학습 중에도 일정 확률로 조건을 제거한 상태(null 입력)로 무조건부 점수를 출력하도록 학습

5. Zero-shot Generalization (무조건 생성 능력 확보)

6. 조건이 주어지지 않은 상태에서도 의미 있는 샘플을 생성할 수 있는 능력 확보
7. 이를 위해 학습 중 일부 데이터에 대해 조건을 제거하고 무조건부 분포에 대한 생성 성능도 함께 학습
8. **파라미터 효율**: 두 개의 네트워크 대신 드롭아웃으로 하나의 모델에 통합

5. 하이퍼파라미터

- **Dropout 확률** p_{drop} **: 보통 0.1~0.2
- **Guidance scale** r **: 1.0(기본)~5.0 이상; 크게 하면 조건 충실도 ↑ 다양성 ↓
- **T (노이즈 스텝 수)**: 모델 리소스 제약에 따라 1000 이상 설정

6. 요약

- **학습**: 조건 드롭아웃으로 단일 네트워크에 조건부/무조건부 예측 모드 학습
- **샘플링**: 매 스텝마다 두 모드를 호출해 $u_{un} + r(u_{cond} - u_{un})$ 로 가중합
- **이점**: 추가 classifier 없이 조건 강화 구현, 파라미터·메모리 효율, 유연한 생성 제어

End of Note