# DALL-E: Zero-Shot Text-to-Image Generation

https://youtu.be/B6DqhOVA4TQ

DALL-E란

DALL-E 동작 과정

DALL-E 2

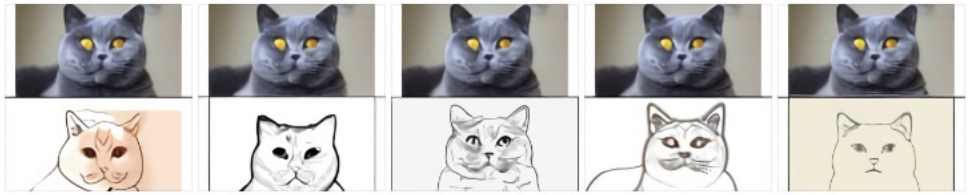
# DALL-E란

- **120억 개의 파라미터**와 **2.5억 개의 데이터(text-image 쌍)**으로 학습시킨 **Large-Scale의 Generative Model**
  - **텍스트를 통해 이미지를 생성**하거나, **텍스트-이미지 쌍을 통해 새로운 이미지를 생성**
  - https://openai.com/blog/dall-e/ 에서 직접 테스트 가능

“An illustration of a baby daikon radish in a tutu walking a dog”

“an armchair in the shape of an avocado”


”a store front that has the work ‘openai’ written on it”



“the exact same cat on the top as a sketch on the bottom”

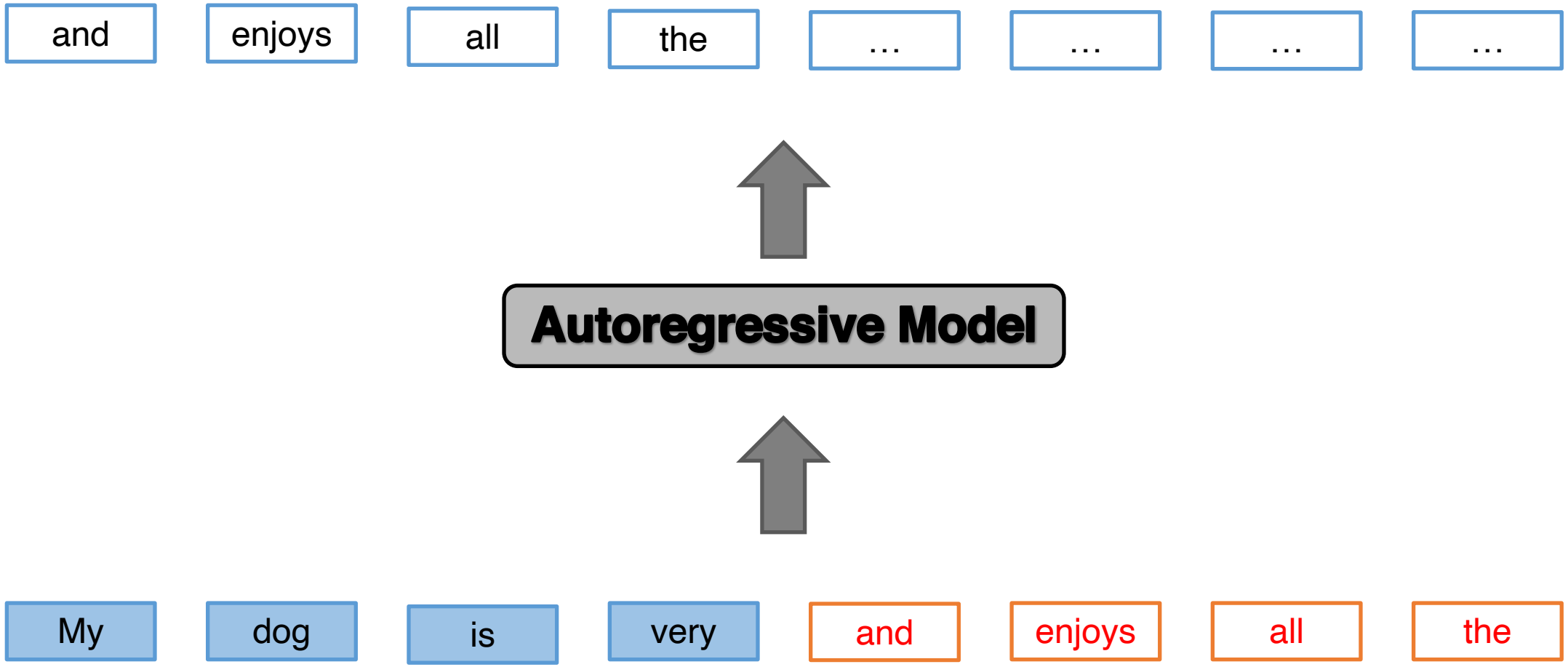
# DALL-E란

## • Contribution

- **2.5억 개의 text-image쌍 데이터**를 통해 **120억 개의 파라미터**를 갖는 **Autoregressive Transformer**를 학습시켰다.
- **자연어**를 통해 **컨트롤할 수 있는** 성능이 뛰어난 **생성 모델**이다.
- **Zero-Shot 상황**에서도 매우 뛰어난 성능을 보인다.
- **Image-to-Image translation**에서도 뛰어나다.

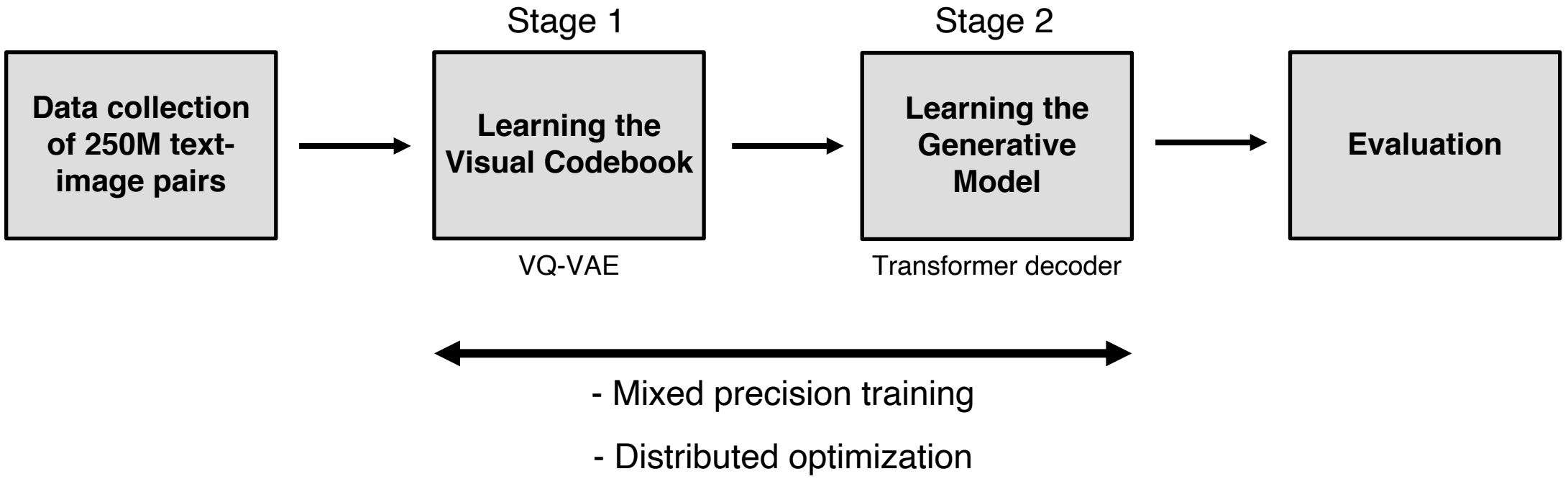
# DALL-E란

- **Autoregressive Model**



# DALL-E 동작 과정

- Overview



# DALL-E 동작 과정

- **Data Collection (2.5억 개)**
  - 이미지 + 캡션
  - YFCC, MS-COCO 이미지
  - Wikipedia의 이미지
- **전처리**
  - 매우 짧은 캡션
  - 영어가 아닌 캡션
  - 가로-세로 비율이 이상한 이미지
  - 등등..

# DALL-E 동작 과정

- **Stage 1 (Learning the Visual Code)**
  - 기존의 이미지는 256x256의 크기로 65,536개의 픽셀을 가짐
  - VQ-VAE를 사용하여 32x32의 latent representation로 변환

- **Vector Quatization (VQ)란?**
  - 특징 벡터를 특징 벡터 집합에 매핑하는 것
  - Ex)
    - 특징 벡터 = {유재석, 지드래곤, 이정재, 싸이, 아이유, 마동석, 강호동}
    - 특징 벡터 집합 = {가수, 영화배우, 개그맨}
      - 가수 = {지드래곤, 싸이, 아이유}
      - 영화배우 = {이정재, 마동석}
      - 개그맨 = {유재석, 강호동}

# DALL-E 동작 과정

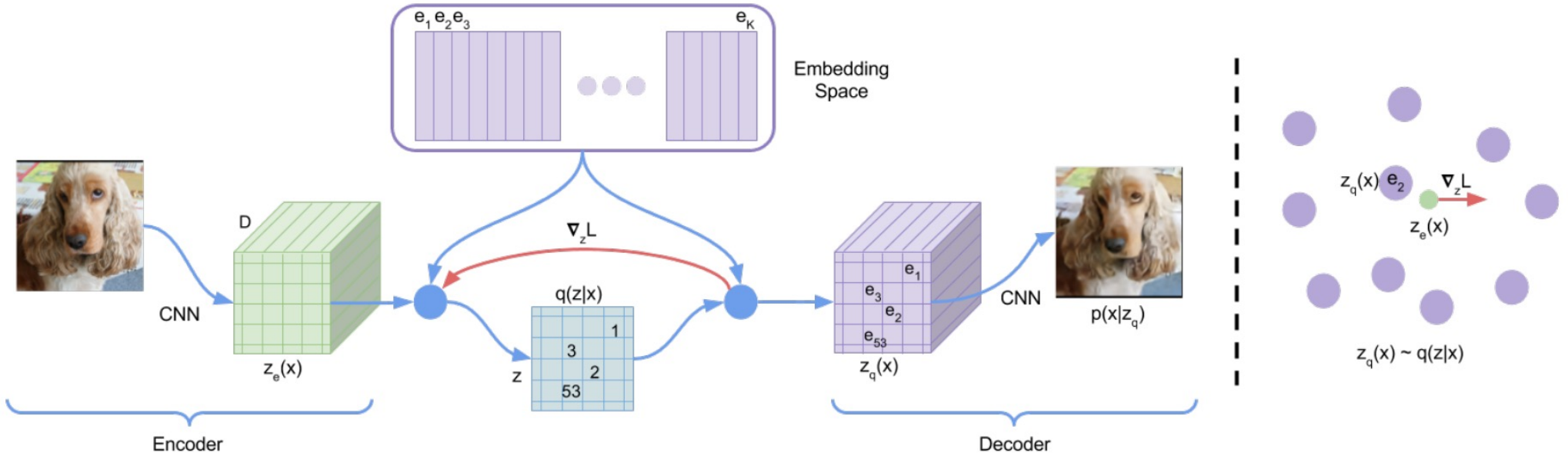
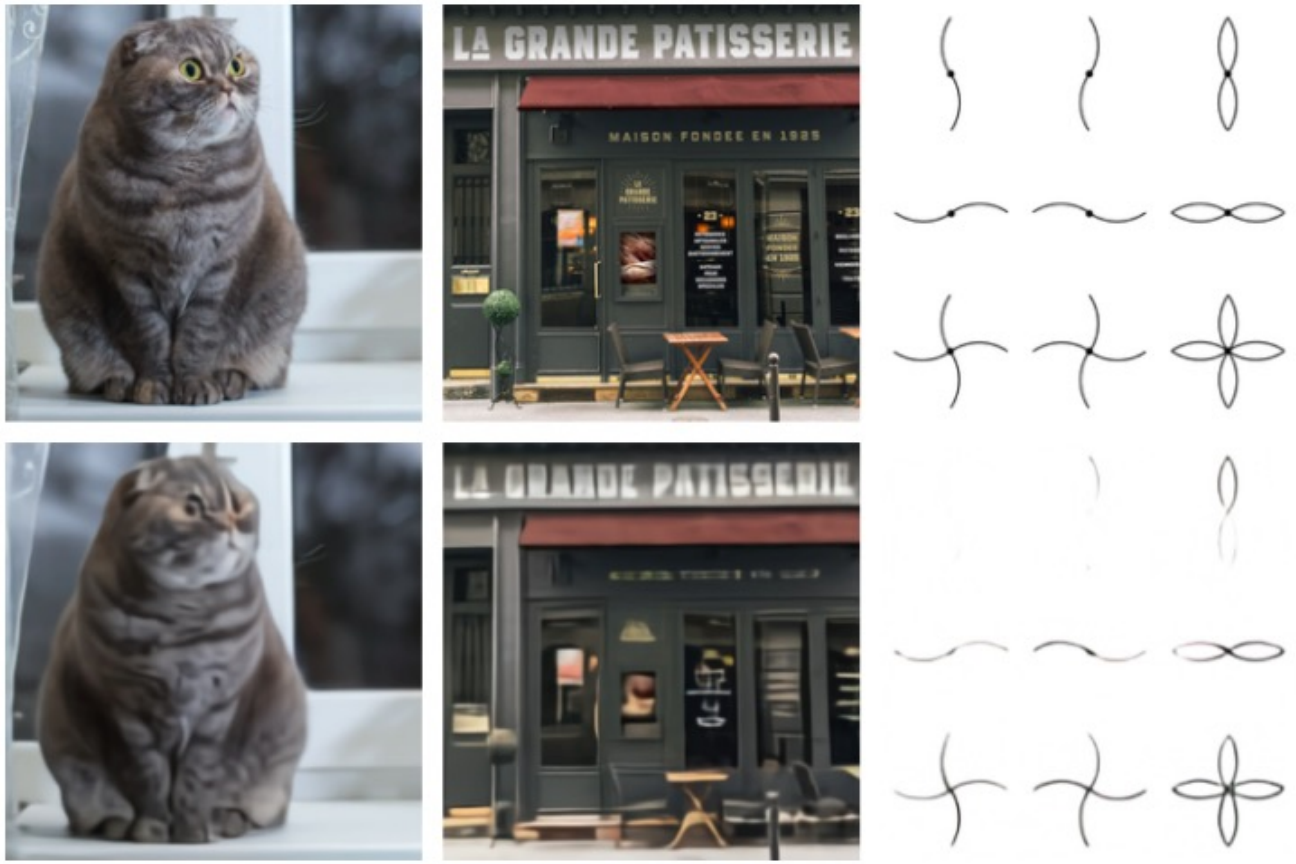



Figure 1: Left: A figure describing the VQ-VAE. Right: Visualisation of the embedding space. The output of the encoder $z(x)$ is mapped to the nearest point $e_2$. The gradient $\nabla_z L$ (in red) will push the encoder to change its output, which could alter the configuration in the next forward pass.

# DALL-E 동작 과정

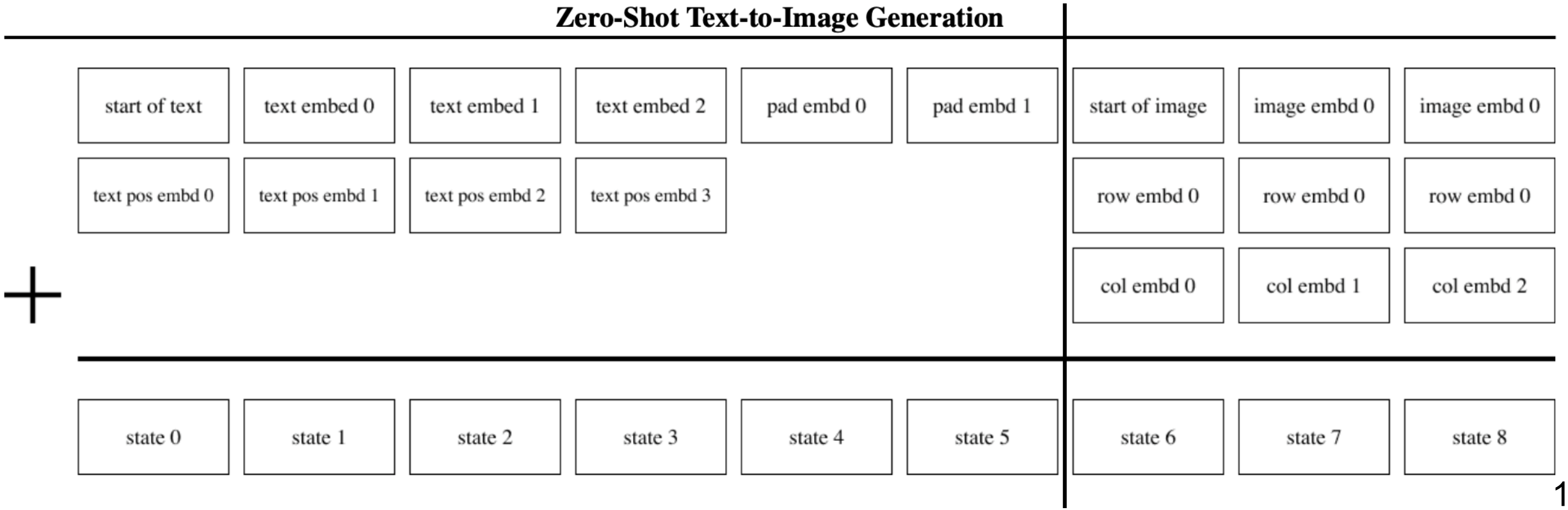
- **VQ**를 통해 **latent representation**을 얻을 수 있음
- 약간의 **blur 현상**이 발생

  → detail한 특징은 감소하더라도, 전체적인 맥락은 유지하면서 효율적인 학습이 가능하도록 한다.



# DALL-E 동작 과정

- 결과적으로 이미지는 **codebook**을 통해 **32x32의 이미지 토큰**으로 변경
- 텍스트(캡션)는 **256개의 텍스트 토큰**으로 변경
- **텍스트 + 이미지의 데이터**를 **Autoregressive Transformer**를 통해 학습



# DALL-E 동작 과정

- **Mixed precision training**
  - 실수는 개수가 무한하므로 컴퓨터상에서 표현하기에 한계가 있음
  - 따라서, 비트를 통해 실수를 표현하기 위한 부동소수점(Floating Point)라는 개념이 존재
  - 32-bit를 사용하여 실수를 표현하는 Single Precision
  - **16-bit(Half Precision)**, 64-bit(Double Precision), 128-bit(Quadruple Precision)
  - 16-bit(Half Precision): 비교적 **정확도 측면**에서 **성능이 떨어지지만, 속도 측면**에서는 **성능이 뛰어남**
  - 32-bit(Single Precision): 비교적 **속도 측면**에서는 **성능이 떨어지지만 정확도 측면**에서는 **성능이 뛰어남**

→ 32-bit Precision의 **정확도에서의 이점**과 16-bit Precision의 **속도에서의 이점**을 갖는 것이 Mixed Precision

→ 파라미터, Adam moments, activation은 16-bit Precision에 저장된다.

# DALL-E 동작 과정

- **Distributed Optimization**
  - 모델이 매우 크기때문에(12억 개의 파라미터) **하나의 gpu에 올라가지 않음**(약 50GB의 메모리 소모)
  - 모델을 쪼개서 **여러개의 gpu를 통해 학습 진행**
  - DALL-E는 1024의 batch size를 가지며, 1024개의 gpu를 통해 학습을 진행
  - **→ gpu당 1개의 배치사이즈 담당**

# DALL-E 2

- **CLIP, Diffusion Model 사용**

”Teddy bears working on new AI research underwater with 1990s technology”

”Teddy bears mixing sparkling chemicals as mad scientists in a steampunk style”

# Q & A