Scalable Diffusion Models with Transformers

Wei Qi Yan

AUT, New Zealand

Table of Contents

Introduction

RNNs

- Most NLP systems relied on gated RNNs, such as LSTMs and gated recurrent units (GRUs), with added attention mechanisms.
- RNNs (LSTM, GRU, etc) have been firmly established approaches in sequence modelling and transduction problems such as language modelling and machine translation.
- RNN models typically factor computations along the symbol positions of the input and output sequences.
- This nature of RNNs precludes parallelisation within training examples.

https://en.wikipedia.org/wiki/Transformer_(machine_learning_model)

Introduction

Transformers

- Transformer is based solely on attention mechanisms, dispensing with recurrence and convolutions entirely.
- Transformers are the current state-of-the-art type of model for dealing with sequences, e.g., in text processing, machine translation, etc.
- Transformers were introduced in 2017 by Google Brain for NLP problems, replacing RNN models (LSTM).
- Transformer models are trained with large datasets.
- Transformer models can be fine-tuned for specific tasks.

https://en.wikipedia.org/wiki/Transformer_(machine_learning_model)

Introduction

Transformers

Transformers can be understood in terms of three components:

- An encoder that encodes an input sequence into state representation vectors.
- A decoder that decodes the state representation vector to generate the target output sequence.
- An attention mechanism that enables our transformer model to focus on the right aspects of the sequential input stream. This is used repeatedly within both the encoder and the decoder to help them contextualise the input data.

https://en.wikipedia.org/wiki/Transformer_(machine_learning_model)

Introduction

Transformers

- Transformer is a deep learning model that adopts the mechanism of self-attention, deferentially weighting the significance of each part of the input data.
- Like RNNs, transformers were designed to handle sequential input data, such as natural languages, for tasks such as *translation* and text *summarisation*.
- Unlike RNNs, transformers do not necessarily process the data in order. The attention mechanism provides context for any position in the input sequence.
- Transformer allows for more parallelisation than RNNs, therefore reduces training times.

https://en.wikipedia.org/wiki/Transformer_(machine_learning_model)

Introduction

Transformers

- Transformers are built on attention mechanisms which can match the performance of RNNs with attention.
- **BERT (Google)**: Bidirectional Encoder Representations from Transformers (BERT) was pre-trained based on two tasks: (1) Language modelling; (2) The next sentence prediction.
- **GPT (OpenAI)**: Generative Pre-trained Transformer (GPT) shows how a generative model of language is able to acquire world knowledge and process long-range dependencies by pre-training on a diverse corpus with long stretches of contiguous text.

https://en.wikipedia.org/wiki/Transformer_(machine_learning_model)

Introduction

GPT

Generative Pre-trained Transformer (GPT):

- **2018**: GPT-1 model is able to acquire world knowledge and process long-range dependencies by pre-training on a diverse corpus with long stretches of contiguous text.
- **2019**: GPT-2 is a general-purpose learner and achieves the state-of-the-art accuracy.
- **2020**: GPT-3 succeeds at meta-learning tasks, which can generalize the purpose of a single input-output pair.
- **2022**: ChatGPT is built on OpenAI's GPT-3.5 language models which has been fine-tuned (an approach to transfer learning) using both supervised and reinforcement learning.
- **2023**: GPT-4 is capable of accepting text or image inputs, which can also read, analyze or generate up to 25,000 words of text, and write code in all major programming languages.

ChatGPT

ChatGPT features:

- Language understanding
- Text generation
- Large vocabulary
- Multilingual support
- Contextual understanding
- Personalisation
- Multi-turn conversation
- Knowledge retrieval

https://en.wikipedia.org/wiki/ChatGPT

GPT-4

Generative Pre-trained Transformer (GPT-4):

- Pre-trained to predict the next token using both public data and "data licensed from third-party providers";
- Fine-tuned with reinforcement learning from human feedback;
- Substantially increase the parameters from GPT-3;
- Improved "security controls" compared to other AI models;
- More reliable and creative to handle much more nuanced instructions;
- Showed impressive improvements in accuracy;
- Gained the ability to summarize and comment on images;
- Summarized complicated texts, passed a bar exam and several standardized tests.

https://en.wikipedia.org/wiki/GPT-4

Introduction

Transformers

Generative Pre-trained Transformer (GPT-4 Turbo):

- OpenAI announced the GPT-4 Turbo and GPT-4 Turbo with Vision model on November 2023.
- GPT-4 Turbo introduces a 128k context window (the equivalent of 300 pages of text in a single prompt)
- GPT-4 Turbo is 3 times cheaper for input tokens and 2 times cheaper for output tokens compared to the original GPT-4 model.
- The maximum number of output tokens for GPT-4 Turbo is 4,096.

https://help.openai.com/en/articles/8555510-gpt-4-turbo

Transformers

Generative Pre-trained Transformer (GPT-4o):

- OpenAI announced the GPT-4o on May 2024.
- The "o" stands for "omni" signifying its broadened abilities.
- GPT-4o a multimodal AI model, which allows for a more natural and integrated user experience within a single system.
- GPT-4o is twice as fast and 50% cheaper to run.
- GPT-4o is boasts a larger context window (128K) for understanding complex instructions.
- GPT-4o offers a more natural and powerful way for users to interact with computers through text, voice, and images.

https://en.wikipedia.org/wiki/GPT-4o

Computational Methods in GPT Models

- **GPT-1**: Generative Pre-trained Transformer (GPT).
- **GPT-2**: Zero-shot learning (no modifications)
- **GPT-3**: Few-shot learning, Meta learning
- **WebGPT**: Imitation learning, Reward model, Reject sampling
- **InstructGPT**: Reinforcement learning, Proximal Policy Optimization (PPO)
- **GPT-3.5** (ChatGPT): Reward learning, Transfer learning, Reinforcement learning.
- **GPT-4**: Reinforcement learning and human feedback (RLHF), Rule-based reward models

https://en.wikipedia.org/wiki/GPT-4

Datasets in GPT Models

- **GPT-1**: BookCorpus, Common crawl
- **GPT-2**: BookCorpus, WebText, LaMDA, CoQA, CNN and Daily Mail, WMT-14 English-French test set
- **GPT-3**: Common Crawl, WebText2, Books1, Books2, Wikipedia
- **WebGPT**: ELI5, TruthfulQA

https://en.wikipedia.org/wiki/GPT-4

Questions?

Questions?

In OpenAI GPT-4o, the “o” stands for

1. The broadened abilities.
2. The broadened tokens.
3. The broadened ethics.
4. The broadened features.

The right answer is:___

Questions?

OpenAI DALL·E Model

Introduction

- DALL·E models are text-to-image models developed by OpenAI using deep learning methodologies to generate digital images from natural language descriptions, called "prompts."
- DALL·E was revealed by OpenAI in Jan. 2021, and uses a version of GPT-3 model to generate images.
- OpenAI announced DALL·E 2 in Apr. 2022. DALL·E 2 is an AI system that can create realistic images and art from a description in natural language.
- DALL·E 3 was released natively into ChatGPT for ChatGPT Plus in Oct. 2023.

https://en.wikipedia.org/wiki/DALL·E

OpenAI DALL·E Model

Introduction

- DALL·E models are a multimodal implementation of GPT-3 with 12 billion parameters which "swaps text for pixels," trained on text–image pairs from the Internet.
- DALL·E was developed and announced to the public in conjunction with CLIP (Contrastive Language-Image Pre-training).
- CLIP is a separate model based on zero-shot learning that was trained on 400 million pairs of images with text captions scraped from the Internet.
- DALL·E 2 uses a diffusion model conditioned on CLIP image embeddings, which, during inference, are generated from CLIP text embeddings by a prior model.

https://en.wikipedia.org/wiki/DALL·E

CLIP: Contrastive Language-Image Pre-training

- CLIP is a method for training a pair of models. One model takes in a piece of text and outputs a single vector. Another takes in an image and outputs a single vector.
- The loss is the total sum of cross-entropy loss across every column and every row of the matrix $A(v_i, v_j)_{N\times N}$.

$$\mathcal{L} = -\sum_i \ln \frac{e^{v_i \cdot w_i}}{\sum_j e^{v_i \cdot w_j}} - \sum_j \ln \frac{e^{v_j \cdot w_j}}{\sum_i e^{v_i \cdot w_j}}$$

  where N is the size of sample batches. The outputs from the text and image models are w_i and v_i, $i = 1, \cdots, N$.
- The models were trained based on a dataset "WebImageText," containing 400 million pairs of image captions.

https://en.wikipedia.org/wiki/DALL·E

OpenAI DALL·E Model

CLIP: Contrastive Language-Image Pre-training

- DALL·E can "manipulate and rearrange "objects in its images and can correctly place design elements in novel compositions without explicit instruction.
- DALL·E showed the ability to "fill in the blanks" to infer appropriate details without specific prompts.
- DALL·E 2 can "inpainting" and "outpainting" by using context from an image to fill in missing areas using a medium consistent with the original, following a given prompt.
- DALL·E 3 is integrated into ChatGPT Plus.

https://en.wikipedia.org/wiki/DALL·E

OpenAI DALL·E Models

An image generated by DALL·E 3 with GPT-4

"A modern architectural building with large glass windows, situated on a cliff overlooking a serene ocean at sunset"

Questions?

https://en.wikipedia.org/wiki/Transformer_(machine_learning_model)

Diffusion Models

Introduction

- A diffusion model consists of three major components: The forward process, the reverse process, and the sampling procedure.
- The goal of diffusion models is to learn a diffusion process that generates the probability distribution of a given dataset.
- In the latent structure of a dataset, data points diffuse through their latent space.
- The diffusion models can be applied to image denoising, inpainting, superresolution, and image generation.

https://en.wikipedia.org/wiki/Diffusion_model

Diffusion Models

Introduction

- Diffusion models train a neural network to sequentially denoise images blurred with Gaussian noise.
- The model is trained to reverse the process of adding noise to an image.
- After training to convergence, the diffusion models can be used for image generation by starting with an image composed of random noises for the network to iteratively denoise.
- Diffusion models are typically formulated as Markov chains and trained by using variational inference.

https://en.wikipedia.org/wiki/Diffusion_model

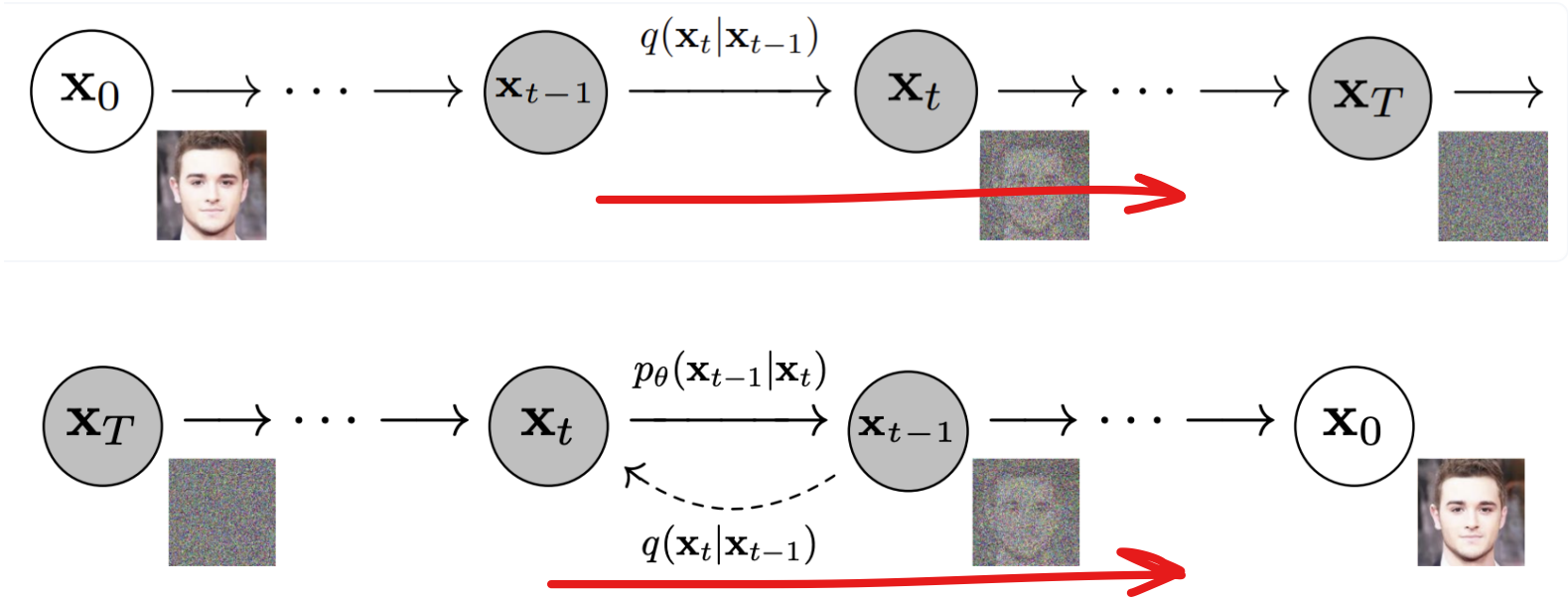
Diffusion Models

Introduction

- Since predicting the noise is the same as predicting the denoised image, then subtracting it from, denoising architecture U-Net was found to be good for denoising images, which is often used for denoising diffusion models that generate images.
- It uses a Transformer network to generate a less noisy trajectory out of a noisy one.
- Diffusion models can be used to perform upscaling.
- Cascading diffusion model stacks multiple diffusion models one after another, in the style of Progressive GAN.
- DALL·E 2 is a cascaded diffusion model that generates images from text by "inverting the CLIP image encoder", termed as "unCLIP".

https://en.wikipedia.org/wiki/Diffusion_model

Denoising Diffusion Probabilistic Models



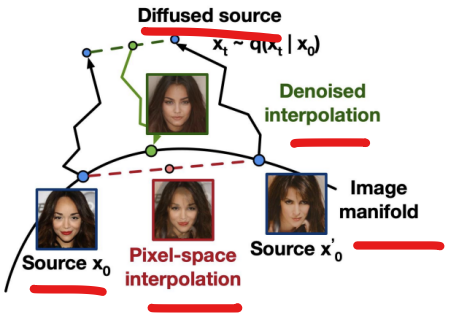
J. Ho, A. Jain, P. Abbeel. Denoising diffusion probabilistic models. NeurIPS 2020.

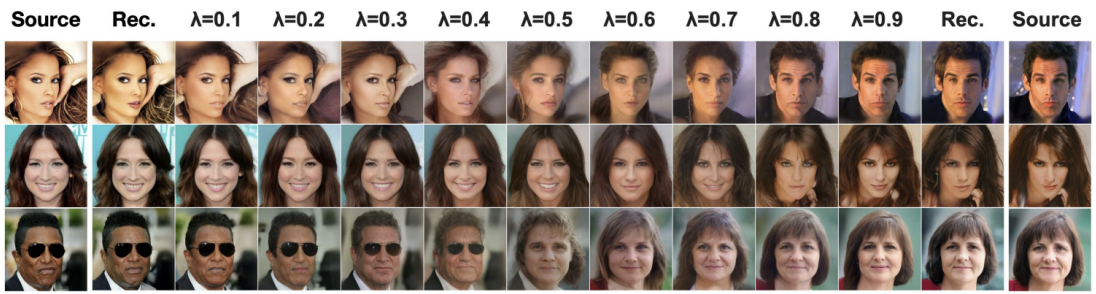
Diffusion Models

Denoising Diffusion Probabilistic Models

J. Ho, A. Jain, P. Abbeel. Denoising diffusion probabilistic models. NeurIPS 2020.

Denoising Diffusion Probabilistic Models





J. Ho, A. Jain, P. Abbeel. Denoising diffusion probabilistic models. NeurIPS 2020.

Questions?

J. Ho, A. Jain, P. Abbeel. Denoising diffusion probabilistic models. NeurIPS 2020.

DiT: Diffusion Transformer

Introduction

- A new class of diffusion models are explored based on the transformer architecture.
- We train latent diffusion models of images, replacing the U-Net backbone with a transformer that operates on latent patches.
- The scalability of Diffusion Transformers (DiTs) is analyzed through the lens of forward pass complexity as measured by Gflops.
- We find that DiTs with higher Gflops—through increased transformer depth/width or increased number of input tokens—consistently have lower FID (Frechet Inception Distance).
- The largest DiT-XL/2 models outperform all prior diffusion models on the class conditional ImageNet 512×512 and 256×256 benchmarks, achieving a FID of 2.27.

DiT: Diffusion Transformer

Introduction

- The diffusion models all adopt a convolutional U-Net architecture as the de facto choice of backbone.
- The diffusion models are well-poised to benefit by inheriting best practices and training recipes from other domains, as well as retaining favorable properties like scalability, robustness and efficiency.
- DiTs adhere to the best practices of Vision Transformers (ViTs), which have been shown to scale more effectively for visual recognition than traditional convolutional networks (e.g., ResNet).
- DiTs are scalable architectures for diffusion models: There is a strong correlation between the network complexity (measured by Gflops) and sample quality (measured by FID).

W. Peebles, S. Xie, Scalable Diffusion Models with Transformers. IEEE ICCV 2023.

DiT: Diffusion Transformer

Introduction

- Transformers have shown remarkable scaling properties under increasing model size, training compute and data in the language domain, as generic autoregressive models and as ViTs.
- Transformers have been trained to autoregressively predict pixels.
- Transformers have also been trained on discrete codebooks as both autoregressive models and masked generative models.
- Transformers have been explored in Denoising Diffusion Probabilistic Models (DDPMs) to synthesize non-spatial data; e.g., to generate CLIP image embeddings in DALL·E 2.

W. Peebles, S. Xie, Scalable Diffusion Models with Transformers. IEEE ICCV 2023.

DDPMs: Denoising Diffusion Probabilistic Models

In order to train diffusion models with a learned reverse process, we train $\varepsilon(\theta)$ with $\mathcal{L}_{simple}$

$$\mathcal{L}_{simple}(\theta) = \|\varepsilon_\theta(x_t) - \varepsilon_t\|_2^2$$

We train $\sum_\theta$ with the full $\mathcal{L}(\theta)$.

$$\mathcal{L}(\theta) = -p(x_0|x_1) + \sum_t \mathcal{D}_{KL}(q^*(x_{t-1}|x_t, x_0)\|p_\theta(x_{t-1}|x_t))$$

Once p_θ is trained, new images can be sampled by initializing $x_{t_{max}} \sim \mathcal{N}(0, \mathbf{I})$ and sampling $x_{t-1} \sim p_\theta(x_{t-1}, x_t)$.

W. Peebles, S. Xie, Scalable Diffusion Models with Transformers. IEEE ICCV 2023.

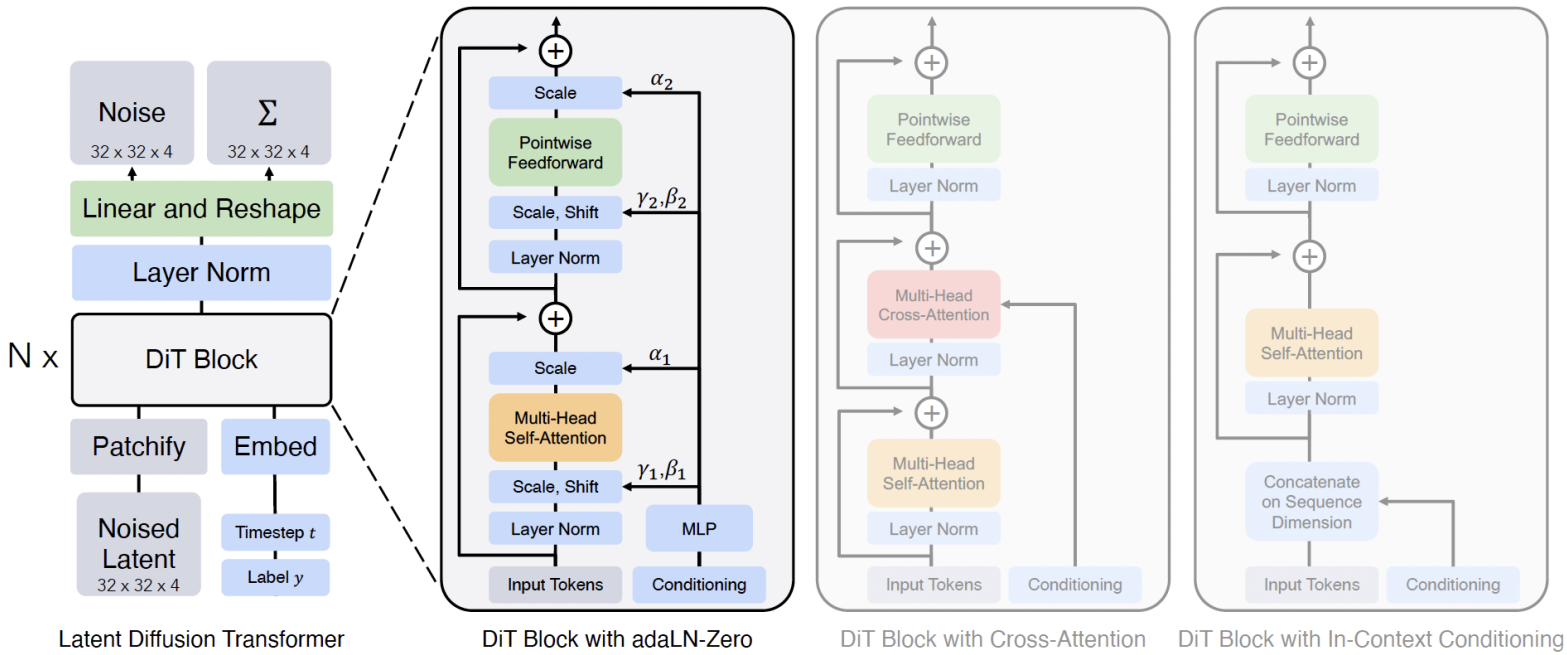
Diffusion Models

LDMs: Latent Diffusion Models

- An autoencoder compresses images x into smaller spatial representations z with an encoder $\mathbf{E}$;
- A diffusion model is trained with the representations $z = \mathbf{E}(x)$.
- New images can then be generated by sampling a representation z and subsequently decoding it to an image $x = \mathbf{D}(z)$.

W. Peebles, S. Xie, Scalable Diffusion Models with Transformers. IEEE ICCV 2023.

The Diffusion Transformer (DiT) architecture



Latent Diffusion Transformer | DiT Block with adaLN-Zero | DiT Block with Cross-Attention | DiT Block with In-Context Conditioning

DiT is based on the Vision Transformer (ViT) architecture which operates on sequences of patches. A smaller patch size results in a longer sequence length.

W. Peebles, S. Xie, Scalable Diffusion Models with Transformers. IEEE ICCV 2023.

The DiT architecture

We include the in-context, cross-attention, adaptive layer norm and adaLN-Zero blocks in the DiTs.

- "Patchify" converts the spatial input into a sequence of tokens. The number of tokens created by patchify is determined by the patch size.
- The input tokens are processed by a sequence of transformer blocks.
- We simply append the vector embeddings as additional tokens in the input sequence, treating them no differently from the image tokens.
- The transformer block is modified to include an additional multi-head cross attention layer following the multi-head self-attention block.
- We explore replacing standard layer norm layers in transformer blocks with adaptive layer norm (adaLN).

The DiT architecture

After the final DiT block, we need decode our sequence of image tokens into an output noise prediction and an output diagonal covariance prediction.

- Both of these outputs have shape equal to the original spatial input.
- We apply the final layer norm (adaptive if using adaLN) and linearly decode each token into a tensor.
- We rearrange the decoded tokens into their original spatial layout to get the predicted noise and covariance.
- The complete DiT design space is patch size, transformer block architecture and model size.
- Scaling the transformer backbone yields better generative models across all model sizes and patch sizes.

W. Peebles, S. Xie, Scalable Diffusion Models with Transformers. IEEE ICCV 2023.

Fréchet distance

The scaling performance is measured by using Fréchet Inception Distance (FID), the standard metric for evaluating generative models of images.

- In mathematics, the Fréchet distance is a measure of similarity between curves that takes into account the location and ordering of the points along the curves.
- FID is a metric used to assess the quality of images created by using a generative model.
- For two multidimensional Gaussian distributions $\mathcal{N}(\mu, \Sigma)$ and $\mathcal{N}(\mu', \Sigma')$,

$$d_F(\mathcal{N}(\mu, \Sigma), \mathcal{N}(\mu', \Sigma')) = \|\mu - \mu'\|^2 + tr(\Sigma + \Sigma' - 2(\Sigma\Sigma')^{\frac{1}{2}})$$

W. Peebles, S. Xie, Scalable Diffusion Models with Transformers. IEEE ICCV 2023.

IS: Inception Score

Inception Score (IS) is an algorithm used to assess the quality of images created by a generative image model.

$$IS(P_{gen}, P_{dis}) = \exp(\mathbf{E}_{x \sim P_{gen}} D_{KL}(P_{gen}, P_{dis}))$$

$$D_{KL}(P_{gen}, P_{dis})) = D_{KL}[P_{dis}(\cdot|x) \| \mathbf{E}_{x \sim P_{gen}} P_{dis}(\cdot|x)]$$

Inception Score only evaluates the distribution of generated images, the FID compares the distribution of generated images with the distribution of a set of real images ("ground truth").

W. Peebles, S. Xie, Scalable Diffusion Models with Transformers. IEEE ICCV 2023.

Conclusion

- Scaling the transformer backbone yields better generative models across all model sizes and patch sizes.
- Increasing model size and decreasing patch size yield considerably improved diffusion models.
- Increasing transformer forward pass Gflops increases sample quality.
- Larger DiT models use large compute more efficiently.
- Scaling both model size and the number of tokens yields notable improvements in visual quality.
- Diffusion Transformers (DiTs) outperform prior U-Net models and inherits the excellent scaling properties of the transformer model class.

W. Peebles, S. Xie, Scalable Diffusion Models with Transformers. IEEE ICCV 2023.

Questions?

Questions?

DiT is based on the Vision Transformer (ViT) architecture which operates on sequences of patches.

1. A smaller patch size results in a longer sequence length.
2. A smaller patch size results in a shorter sequence length.
3. A smaller patch size results in an invariant sequence length.
4. None of the given options.

The right answer is:___

Questions?