HKU Musketeers Foundation Institute of Data Science 香港大學同心基金數據科學研究院

# DATA8018: Deep Generative Models

*https://hku-data8018.github.io/*

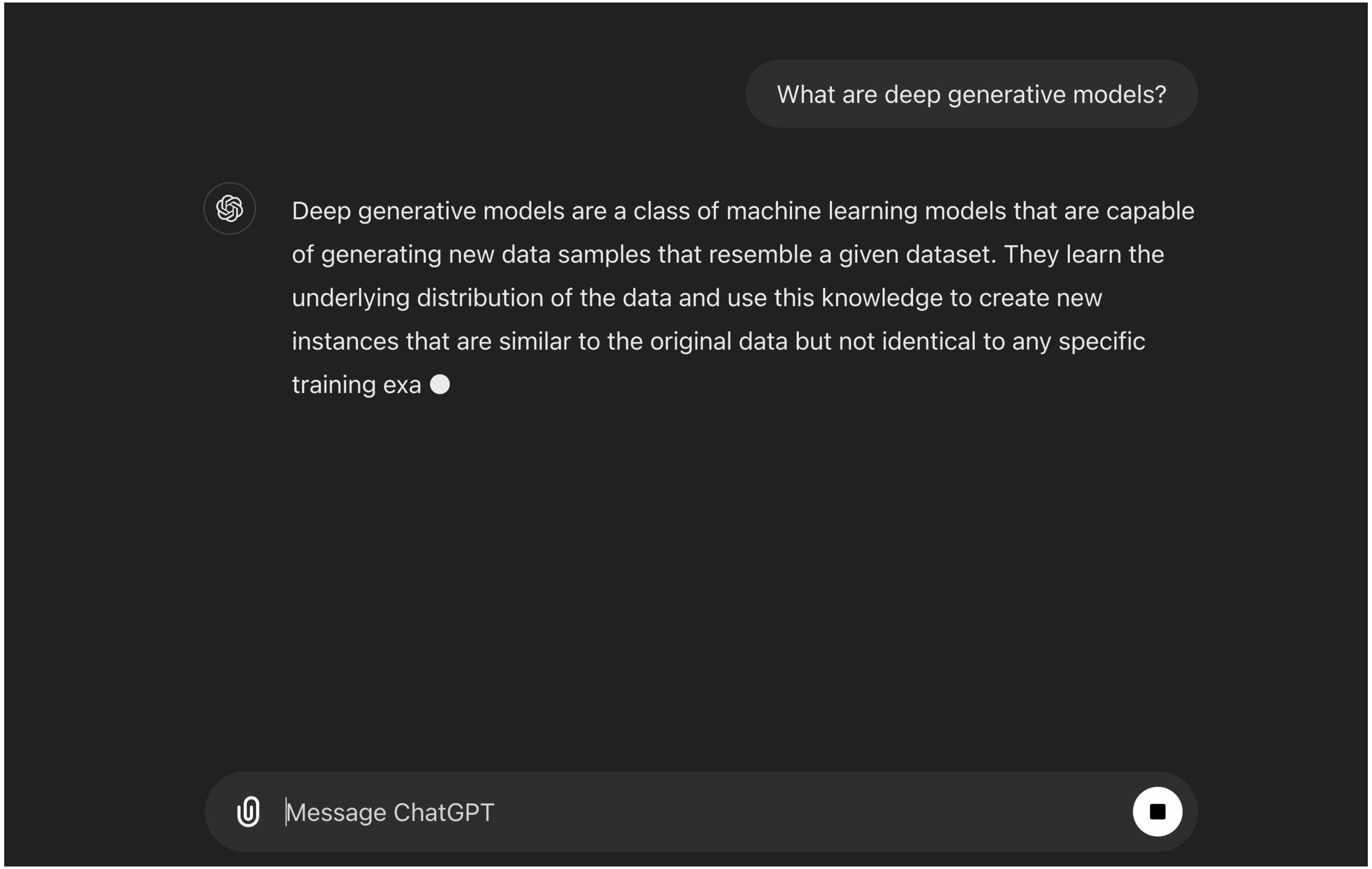
Bo Dai

Spring 2026

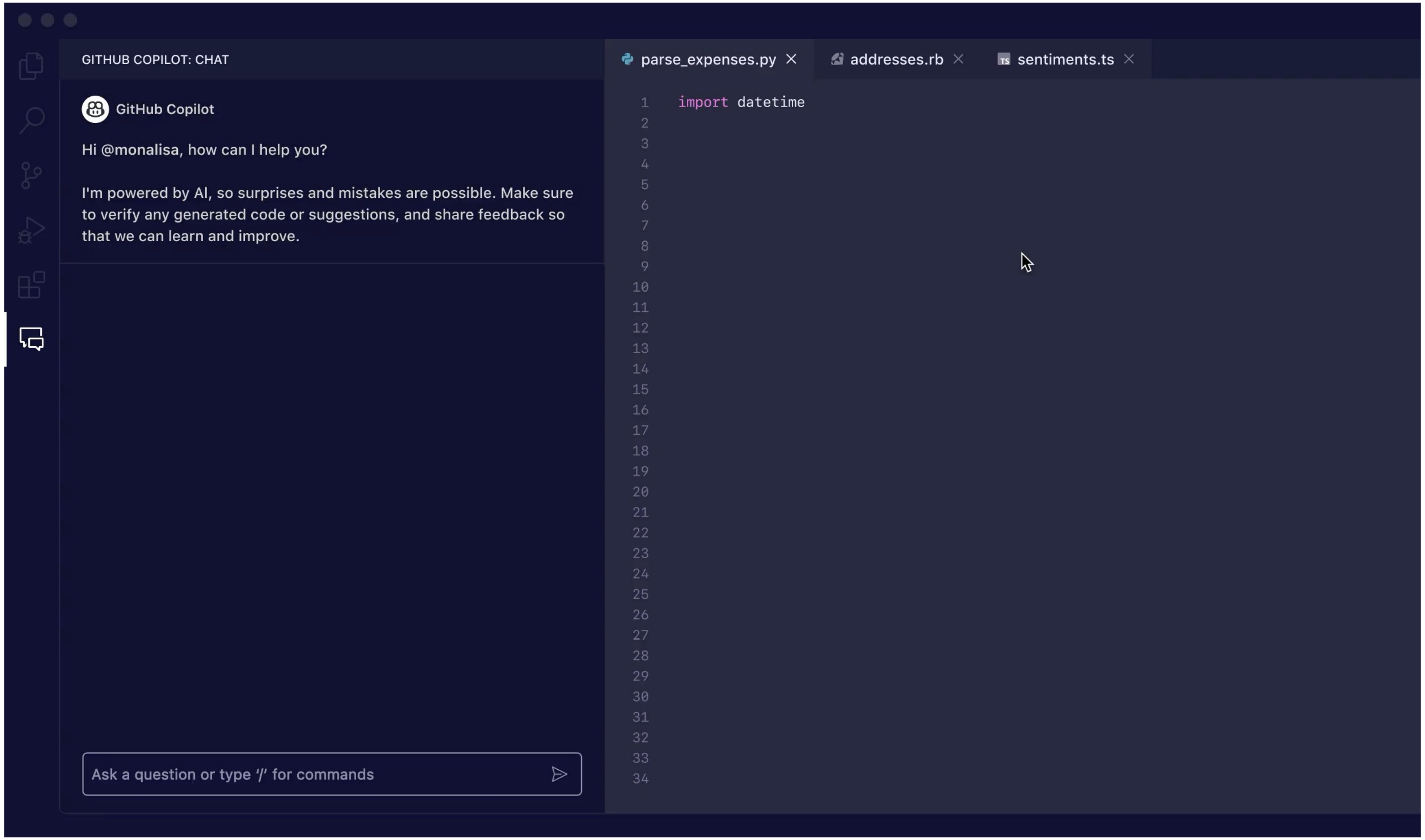
- Slides reference
  - MIT EECS Fall 2024, 6.S978 Deep Generative Models, https://mit-6s978.github.io/

# Era of Generative AI






Language Generation

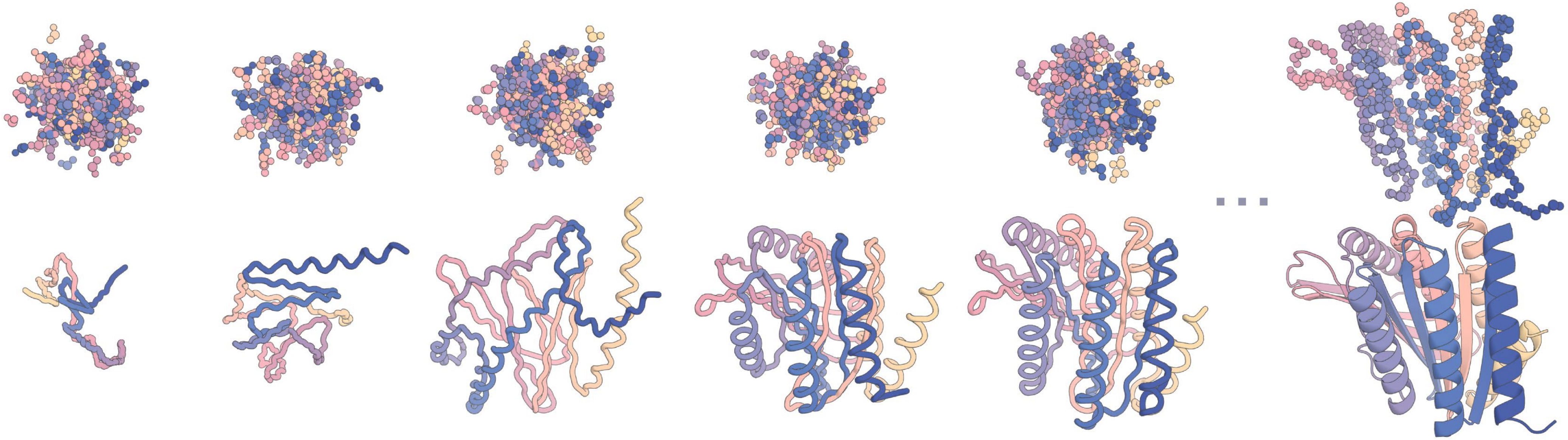



Coding Assistant

# Era of Generative AI

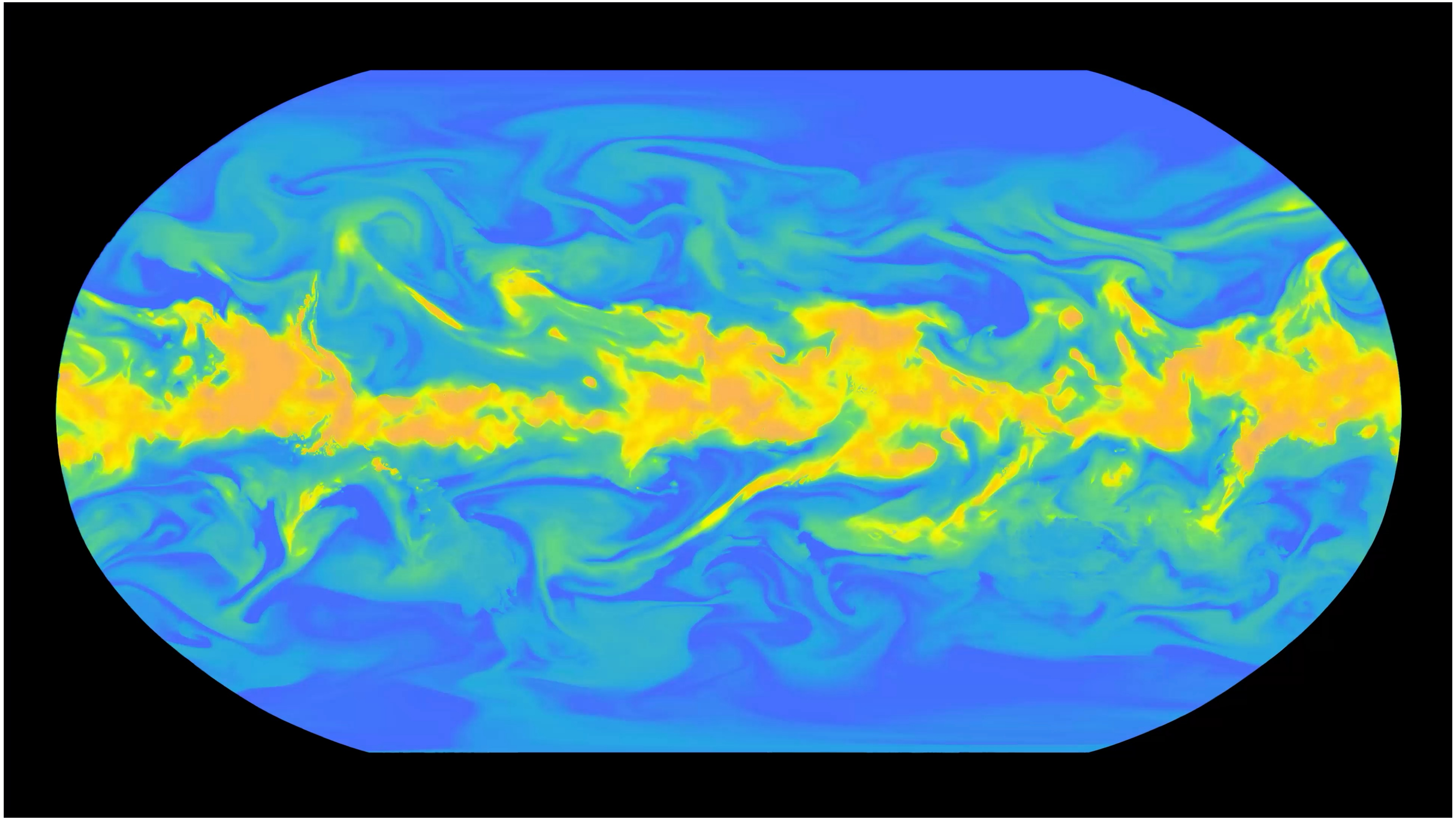

Image Generation


Video Generation



Protein Design

https://www.nature.com/articles/s41586-023-06415-8



Weather Forecasting

https://deepmind.google/science/weathernext/

The 1956 Dartmouth workshop is often regarded as the Birth of Artificial Intelligence as a distinct field.

- Conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. Attempts will be made to solve

# Dartmouth workshop

Article Talk

From Wikipedia, the free encyclopedia

*This article is about conferences related to artificial intelligence. For the peace process conference*

This article's **tone or style may not reflect the encyclopedic tone** used on guide to writing better articles for suggestions. *(June 2018) (Learn how and wh message)*

The **Dartmouth Summer Research Project on Artificial Intelligence** was a 1956 summer workshop widely considered[1][2][3] to be the founding event of artificial intelligence as a field.

The project lasted approximately six to eight weeks and was essentially an extended brainstorming session. Eleven mathematicians and scientists originally planned to attend; not all of them attended, but more than ten others came for short times.

The 1956 Dartmouth workshop is often regarded as the Birth of Artificial Intelligence as a distinct field.

- Conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. Attempts will be made to solve

7) Randomness and Creativity

A fairly attractive and yet clearly incomplete conjecture is that the difference between creative thinking and unimaginative competent thinking lies in the injection of a some randomness. The randomness must be guided by intuition to be efficient. In other words, the educated guess on the hunch include controlled randomness in otherwise orderly thinking.

# History of Generative AI



AIW1

**1642**
First mechanical calculating machine by Blaise Pascal

**1837**
First programmable machine design by Charles Babbage & Ada Lovelace

**1943**
Neural networks foundations by Warren McCulloch & Walter Pitts

**1950**
**TURING TEST**
First method to evaluate a machine's intelligence by Alan Turing (& Three Laws of Robotics by Isaac Asimov)

**1956**
**AI BORN**
"Artificial intelligence" coined by computer scientist John McCarthy

**1965**
**ELIZA**
First chatbot invented by Joseph Weizenbaum

**1969**
**SHAKEY**
First "electronic person," a robot developed by SRI International

**1974–1980**
**AI WINTER**
Many dead-ends for two decades, large decrease in AI research (e.g., DARPA funding cuts)

**1980s**
**NEW HOPE**
First expert systems & commercial applications

AIW2

**1987–1993**
**AI WINTER**
Various setbacks (e.g., with expert systems)

**1997**
**DEEP BLUE**
First chess-playing computer by IBM defeats world champion Garry Kasparov; same goes with Othello

**2007**
**AI CARS**
DARPA Urban Challenge for autonomous cars

**2011**
**SIRI**
Voice recognition on iPhone

**2011**
**WATSON**
IBM's question answering computer Watson wins first place on *Jeopardy!*

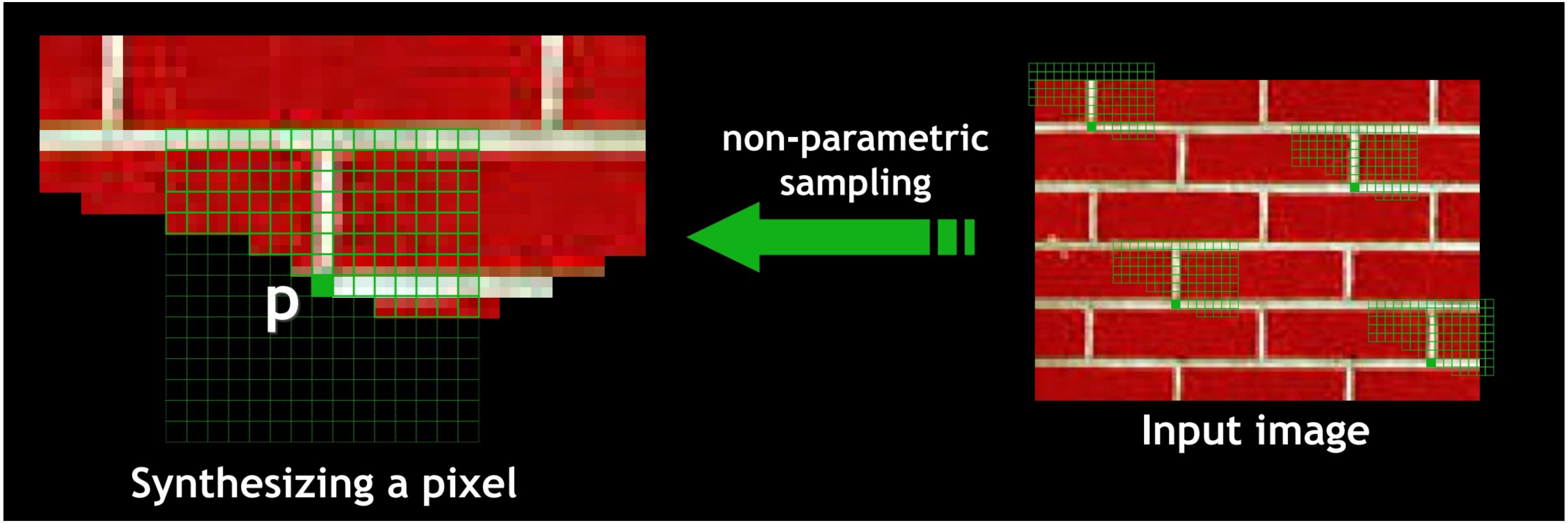
**2014**
**GAN**
First generative adversarial networks

**2015**
**OPEN AI**
Creation of OpenAI by Sam Altman, Elon Musk & others

**2017**
**ALPHA GO**
Google AlphaGo defeats *Go* world champion Ke Jie; AlphaGo Zero trained a GenAI itself

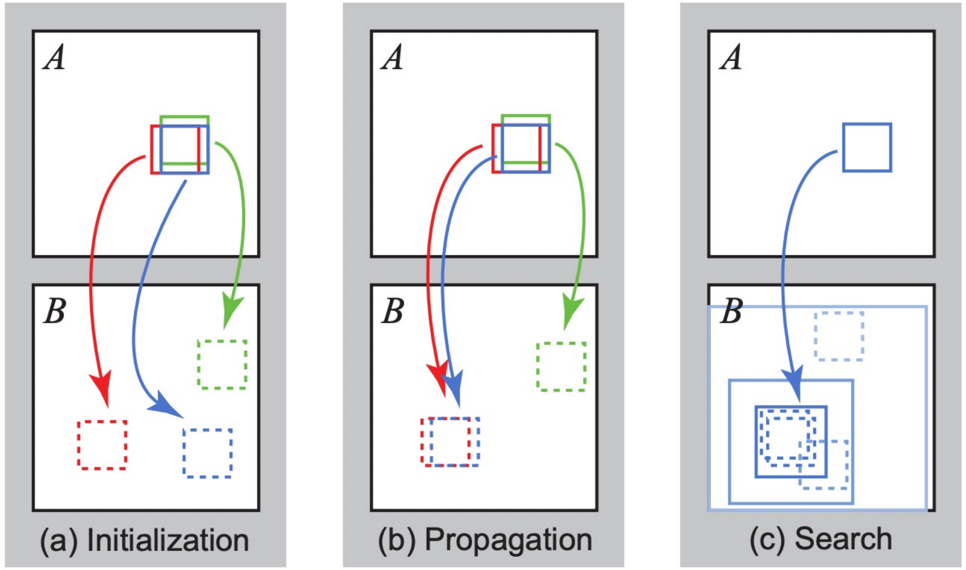
**2020**
**T-NLG & GPT-3**
Introduction of Microsoft T-NLG & OpenAI GPT-3

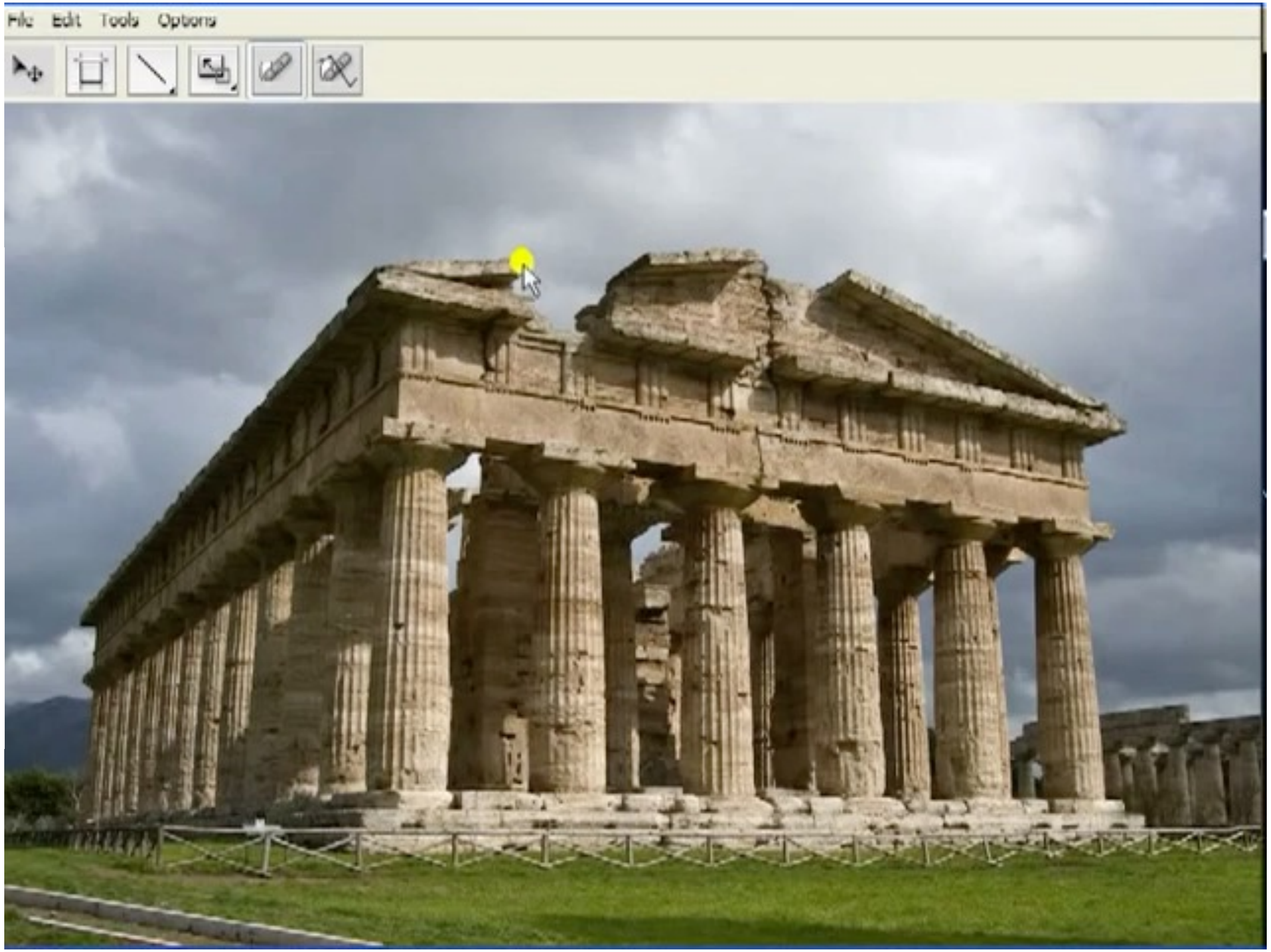
**2022**
**ChatGPT**
Introduction of OpenAI ChatGPT




Texture Synthesis by Non-parametric Sampling, ICCV 1999

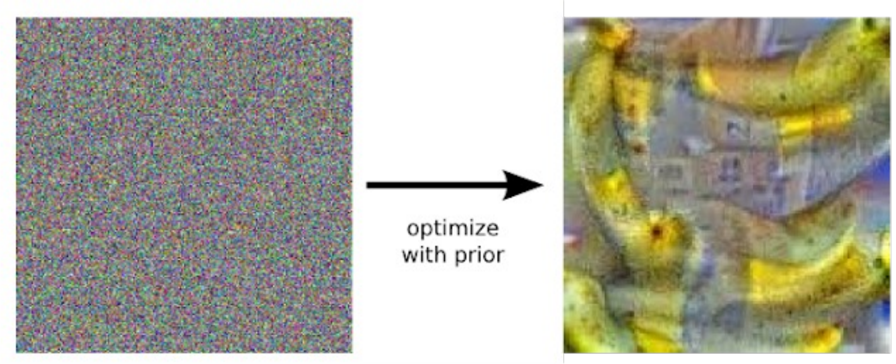
https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/papers/efros-iccv99.pdf

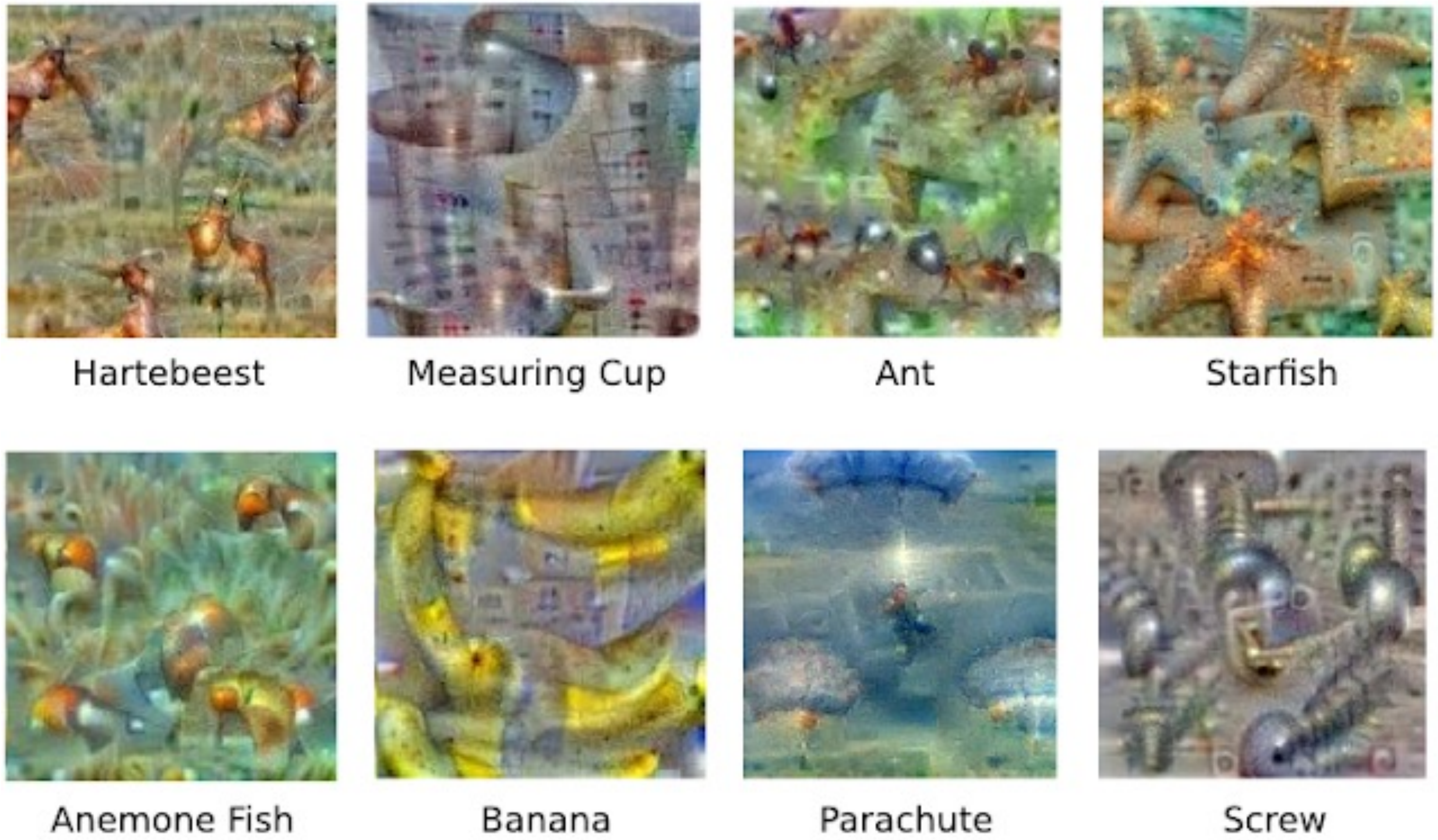



PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing, SIGGRAPH 2009

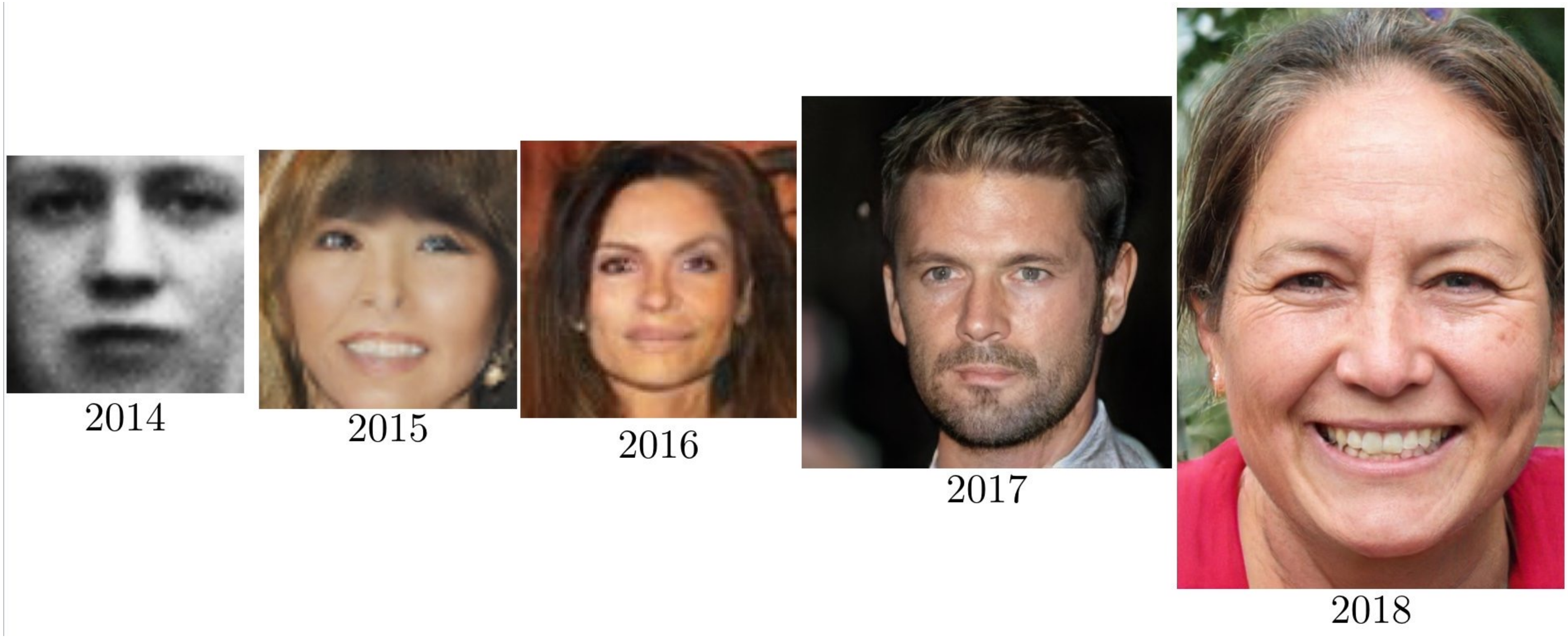
https://gfx.cs.princeton.edu/pubs/Barnes_2009_PAR/patchmatch.pdf



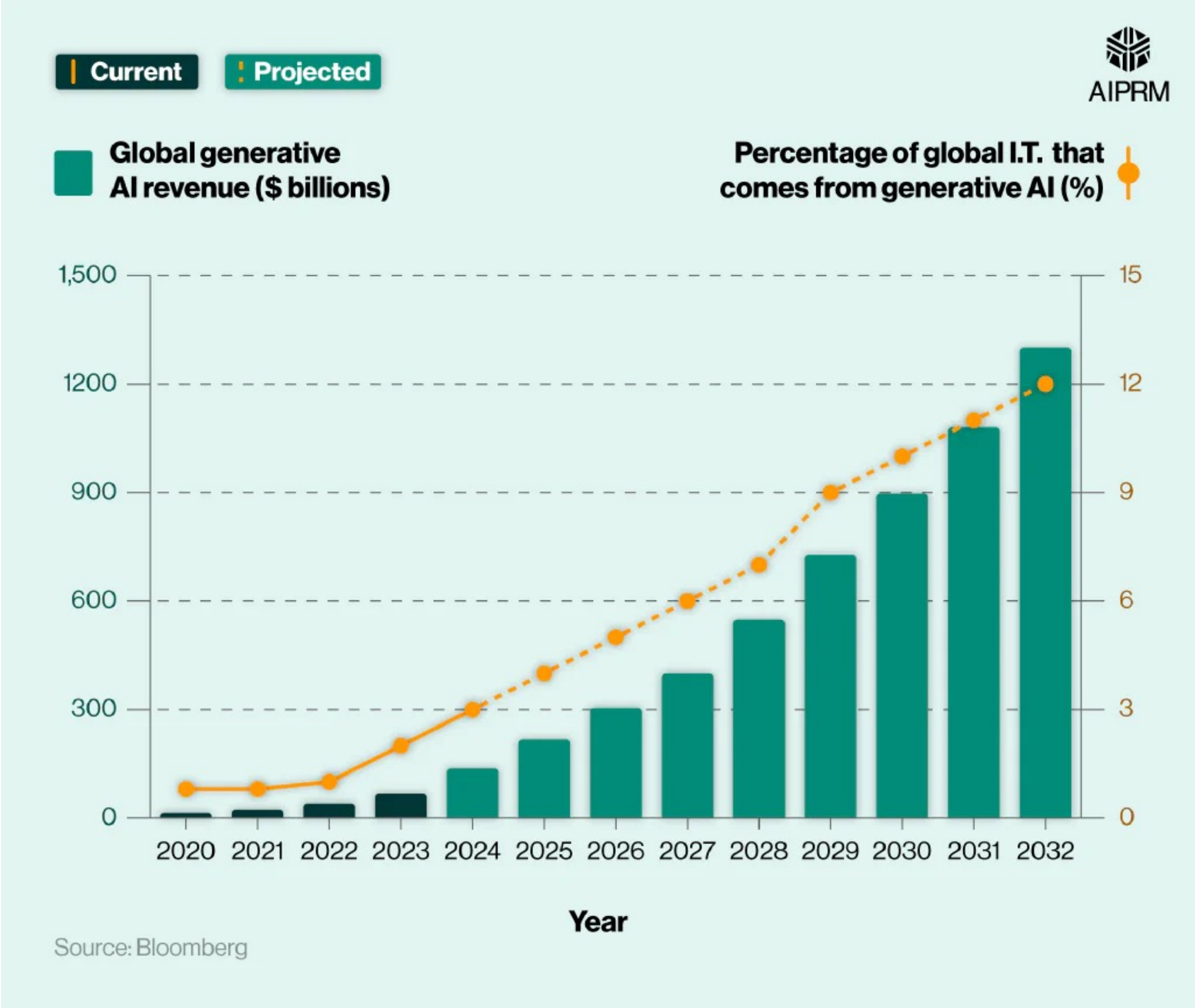

DeepDream, 2015

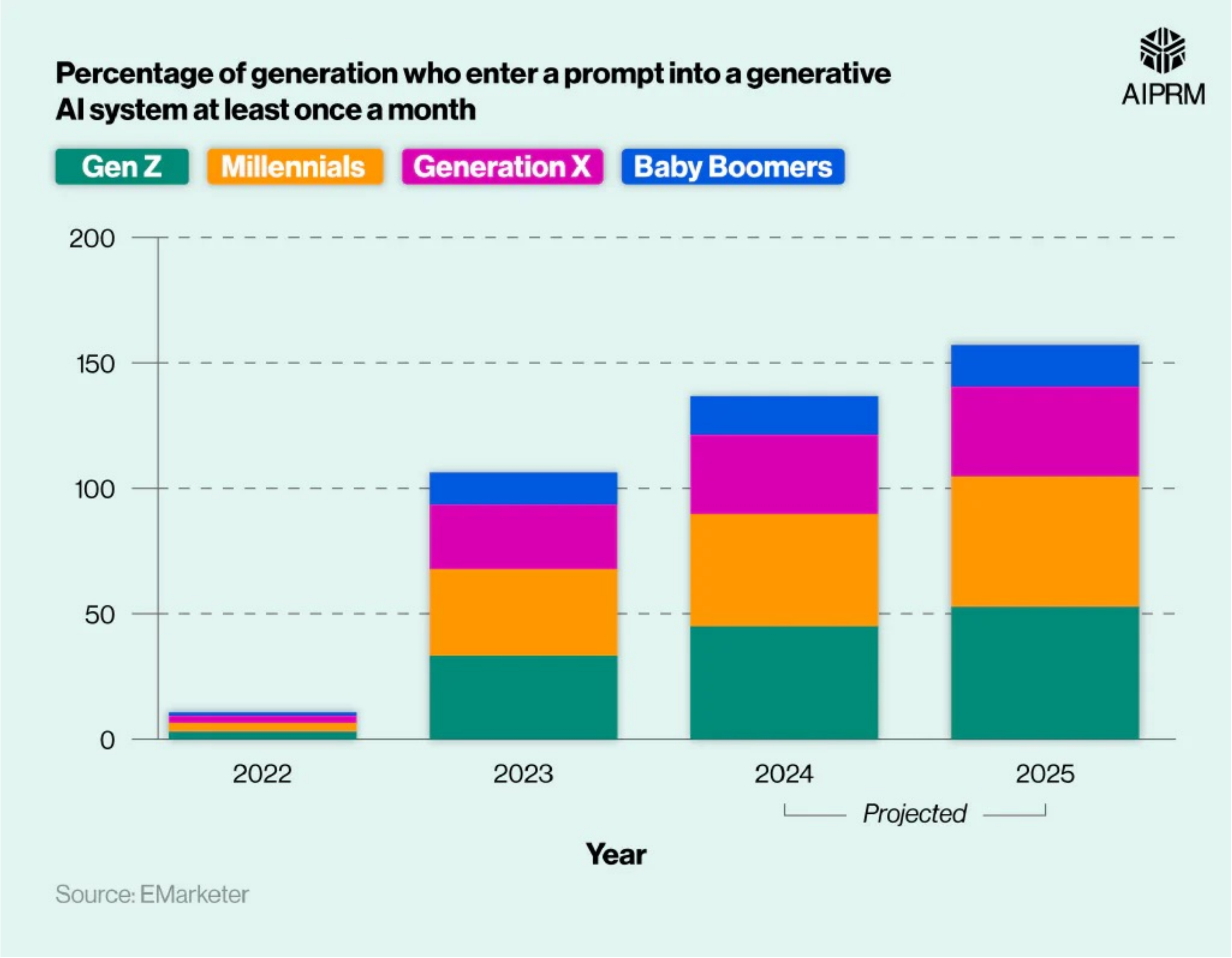
https://research.google/blog/inceptionism-going-deeper-into-neural-networks/




https://colab.research.google.com/drive/1vouPuAYpVkc1Yo4y-ptPNkSMLGJHFRRw#scrollTo=ITZuApL56Mny

# History of Generative AI




https://www.aiprm.com/generative-ai-statistics/

# Course Information



This course will cover:

- Representative generative models and learning algorithms
- How real-world problems are formulated as generative models?
- Challenges, opportunities, open questions

# Course Information

- Grading
  - 50% Paper Reading
    - 10%: register and upload ppt on time
    - 30%: presentation quality
    - 10%: feedback quality
  - 50% Group Project
    - 10%: proposal quality
    - 10%: final presentation
    - 30%: project quality

# Course Information



- Paper Reading
  - 2 for each week, starting at week 3
  - 45 min per student
    - 30~35 min presentation
      - What is the key research question of the paper?
      - How the paper explore this question?
      - Limitation/future work/personal thoughts/…
    - 10~15 min discussion
  - Use provided paper as the anchor, can involve more relevant papers if necessary

- Group Project
  - Maximum of 2 students per group
  - Topic:
    - a literature overview/survey on frontier and specific topic in deep generative models and applications
  - Presentation:
    - 45 min per group, in last two weeks

- Group Project
    - Low quality one:
        - Isolated descriptions of different papers, insufficient details, unclear timeline/trend of advancement in that direction, etc.
    - High quality one:
        - Really useful for someone that is not familiar with that direction
        - Clear demonstration of connections between papers, major paradigms, advancement over time, key challenges, sufficient details, inspiring insights, etc.
        - Predict future research questions
    - Submission:
        - Mid-semester: proposal with bullet points on topics, brief structure, etc.
        - End-semester: a blog in static html page
            - Example:
                - https://lilianweng.github.io/posts/2021-07-11-diffusion-models/
                - https://diffusionflow.github.io/
                - https://spaces.ac.cn/archives/6853
                - https://yang-song.net/blog/2021/score/
                - https://karpathy.github.io/2015/05/21/rnn-effectiveness/
                - https://sander.ai/2023/07/20/perspectives.html
                - https://colah.github.io/posts/2015-08-Understanding-LSTMs/

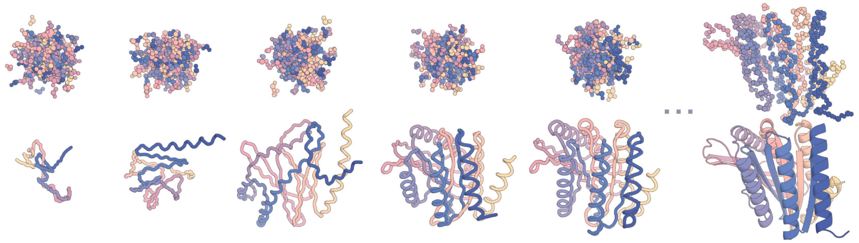
# Course Information

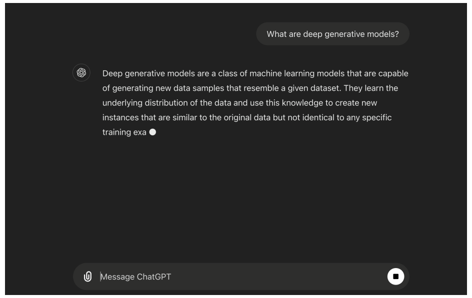
- First come first serve
- Student Presentation Schedule
  - https://hkuhk-my.sharepoint.com/:x:/g/personal/bdai_hku_hk/IQCw7NjfThocQZkrtKkYWOexAUxFssmjnC6OhlxUvSBITEo?e=NbkAHd
- Group Info
  - https://hkuhk-my.sharepoint.com/:x:/g/personal/bdai_hku_hk/IQCCN_t6WAxwR4MGYn4QK_LRAeFud4miJGwrJNAYJWO7kU0?e=FVeeVu
- Final Group Presentation Schedule
  - https://hkuhk-my.sharepoint.com/:x:/g/personal/bdai_hku_hk/IQBpwEA-lzQUSorbZL4MNCi-Ae06e_B-tWlQsj2JxTGxgHo?e=3ntO3G

# What are Generative Models?

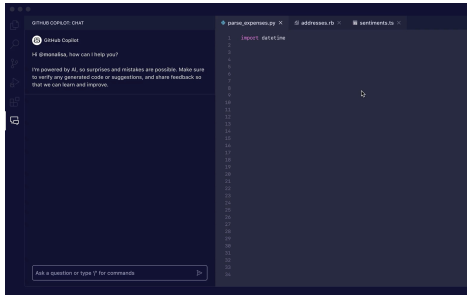
# What are Generative Models?

## What do these scenarios have in common?

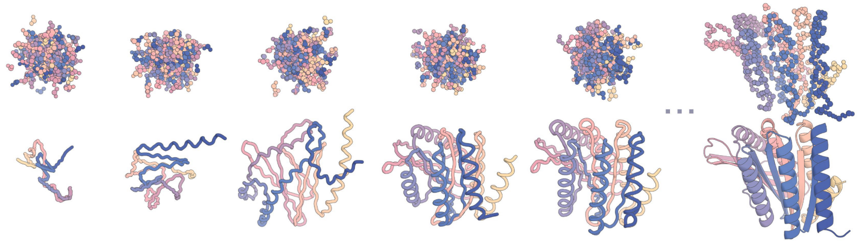


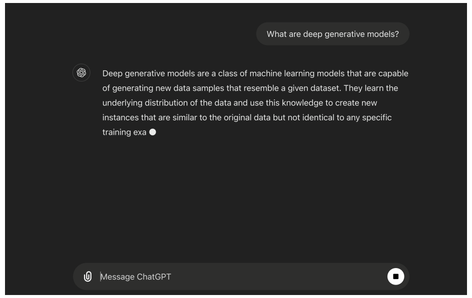





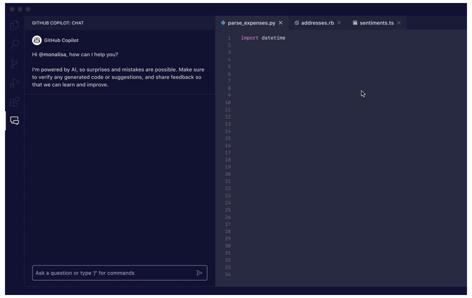


## What do these scenarios have in common?







- There are multiple of infinite valid predictions to one input
- Some predictions are more "plausible" than some others
- Training data may contain no exact solution
- Predictions may be more complex, more informative, and higher dimensional than input

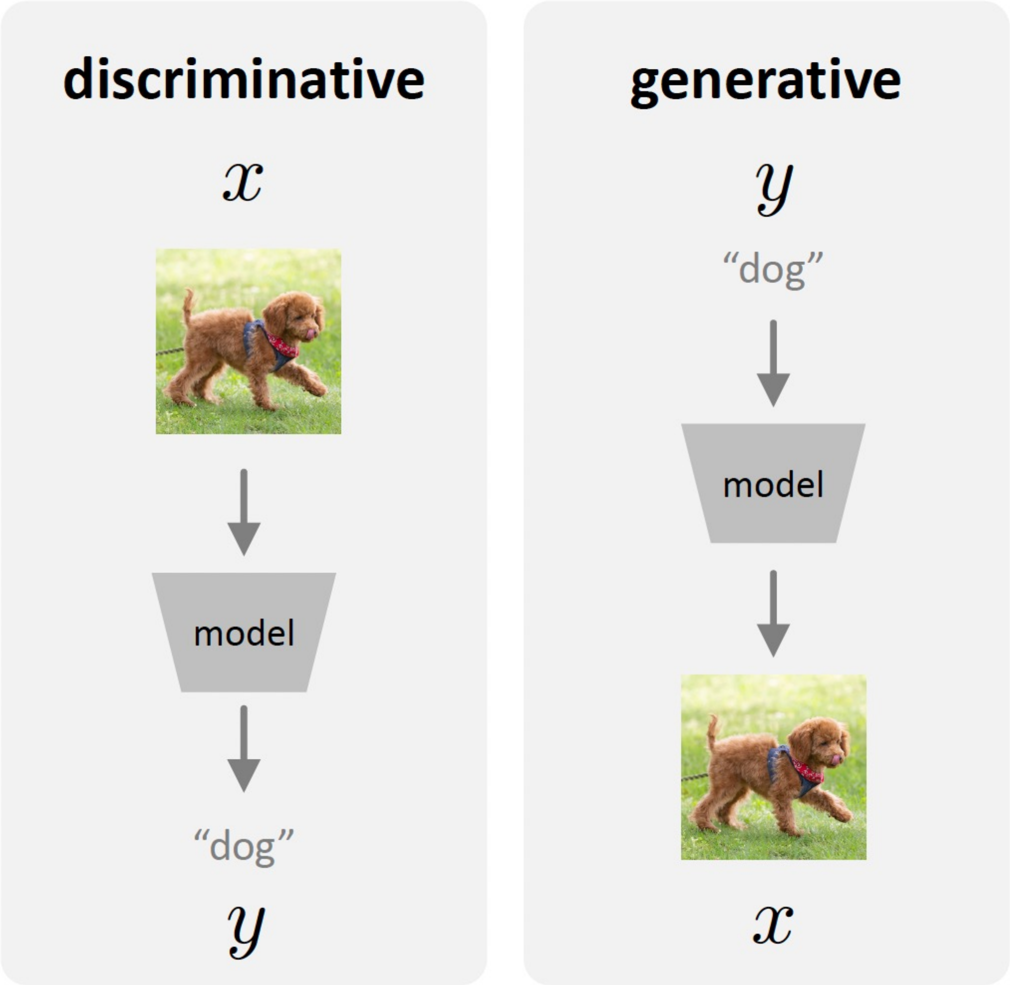
## Discriminative vs. Generative Models

Discriminative

- “sample” x ⇨ “label” y
- one desired output
- concept abstraction

Generative

- “label” y ⇨ “sample” x
- many possible outputs
- detail inflation



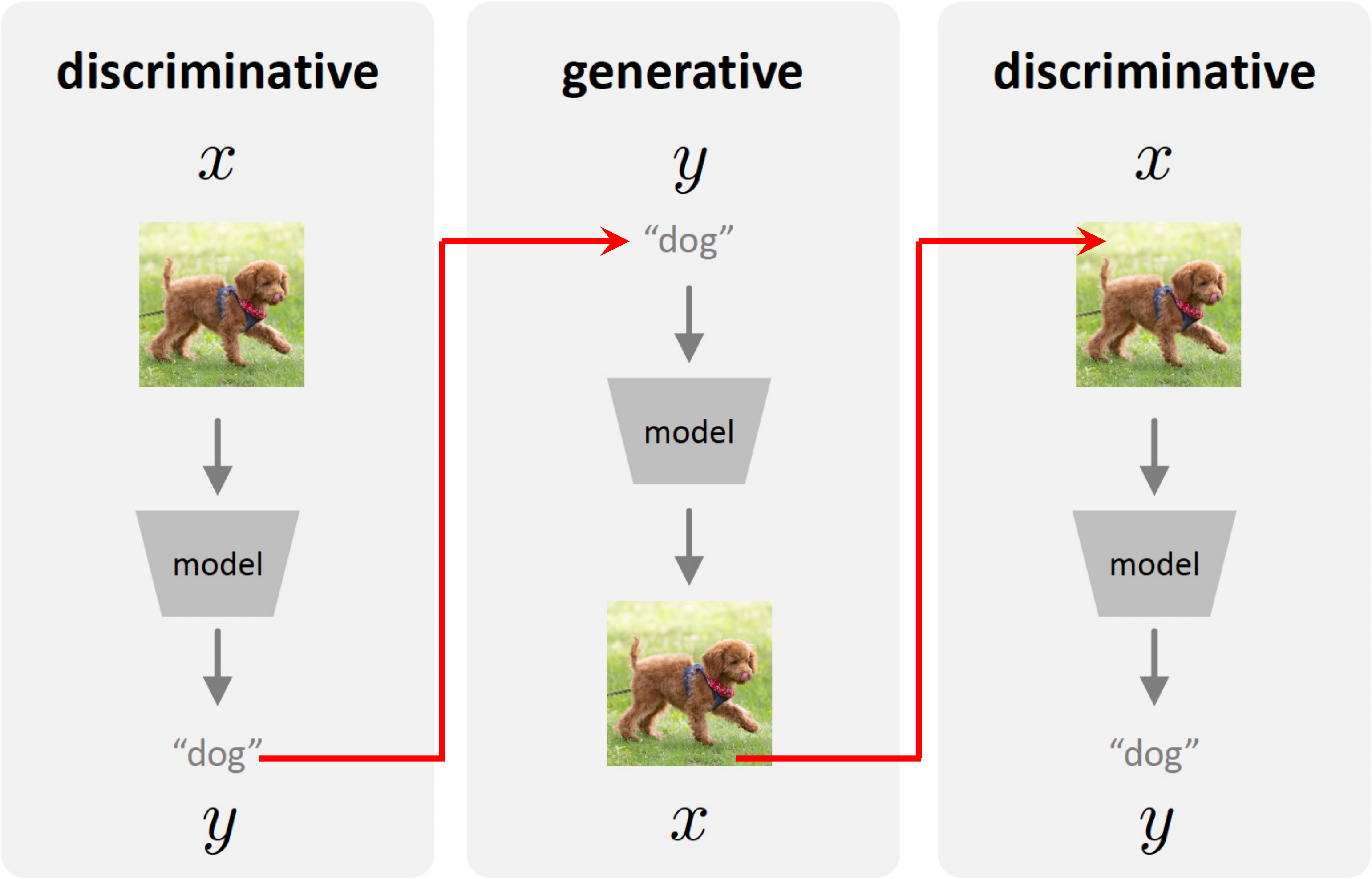
## Discriminative vs. Generative Models

Discriminative

- “sample” x ⇨ “label” y
- one desired output
- concept abstraction

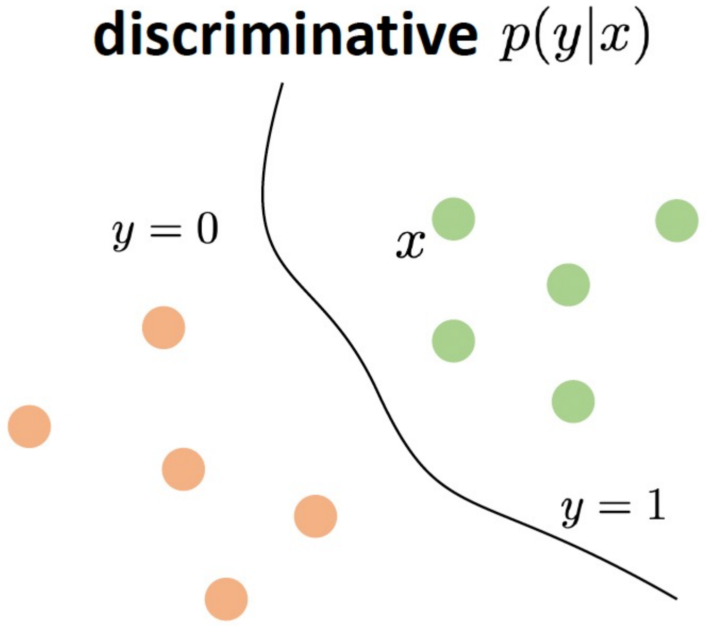
Generative

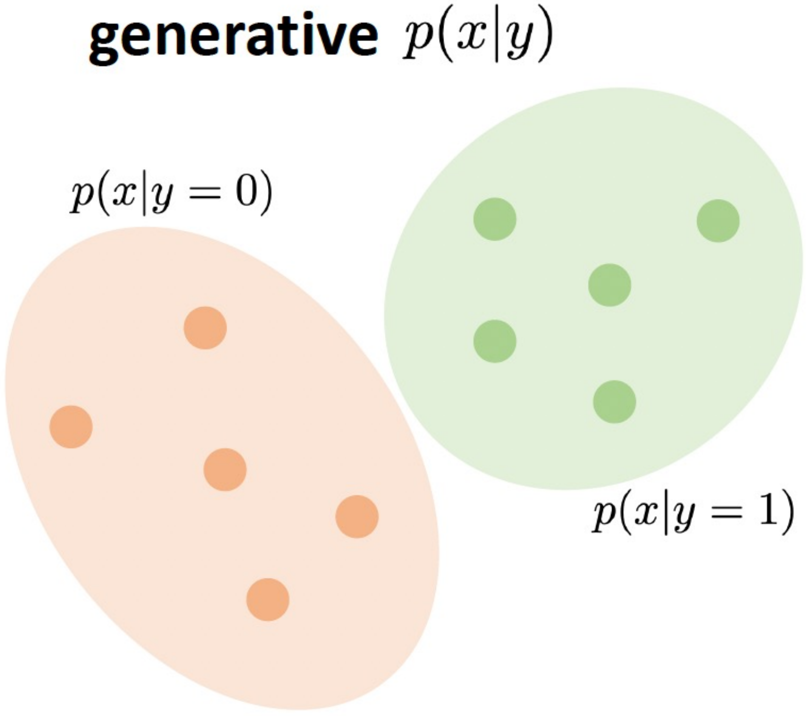
- “label” y ⇨ “sample” x
- many possible outputs
- detail inflation



HKU Musketeers Foundation Institute of Data Science 香港大學同心基金數據科學研究院

## Discriminative vs. Generative Models



## Discriminative vs. Generative Models

Generative models can be discriminative:

$$\underbrace{p(y|x)}_{\text{discriminative}} = \underbrace{p(x|y)}_{\text{generative}} \frac{p(y)}{p(x)}$$

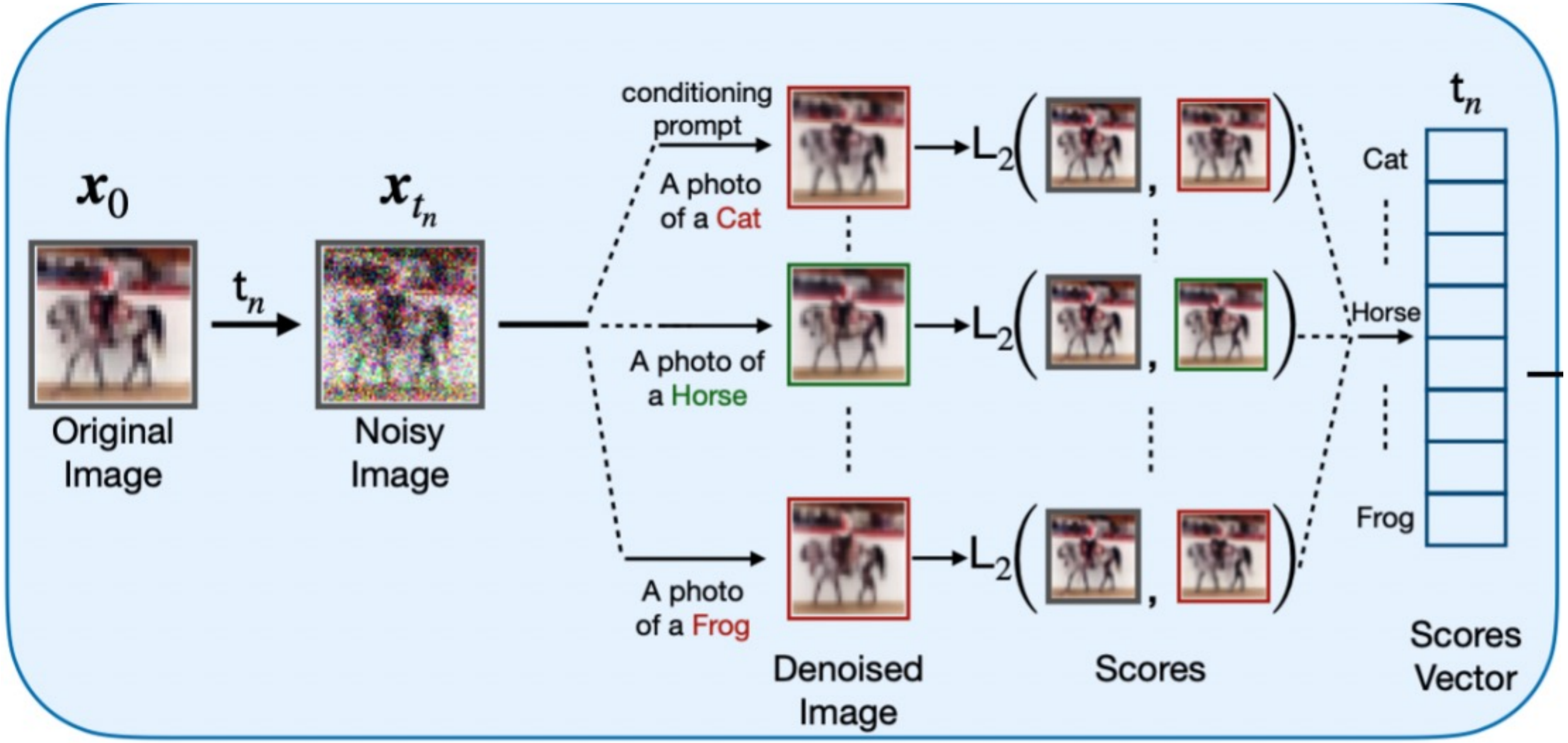
assuming known prior

constant for given x

## Discriminative vs. Generative Models

Generative models can be discriminative:

- In practice




Text-to-Image Diffusion Models are Zero-Shot Classifiers, NeurIPS 2025

https://proceedings.neurips.cc/paper_files/paper/2023/file/b87bdcf963cad3d0b265fcb78ae7d11e-Paper-Conference.pdf

## Discriminative vs. Generative Models

Can discriminative models be generative?

## Discriminative vs. Generative Models

Can discriminative models be generative?

- The challenge is about representing and predicting distributions

$$p(x|y) = p(y|x)\frac{p(x)}{p(y)}$$

still need to model prior distribution of $x$

generative

discriminative

constant for given $y$

## Discriminative vs. Generative Models

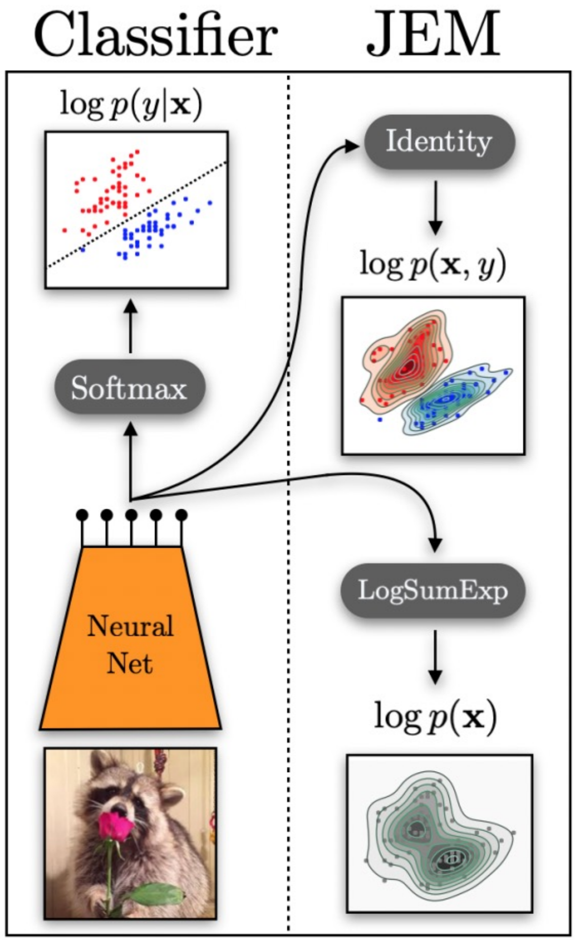
Can discriminative models be generative?

- The challenge is about representing and predicting distributions
- In practice

$$p_\theta(y \mid \mathbf{x}) = \frac{\exp\left(f_\theta(\mathbf{x})[y]\right)}{\sum_{y'} \exp\left(f_\theta(\mathbf{x})[y']\right)}$$

$$p_\theta(\mathbf{x}, y) = \frac{\exp\left(f_\theta(\mathbf{x})[y]\right)}{Z(\theta)}$$

$$p_\theta(\mathbf{x}) = \sum_y p_\theta(\mathbf{x}, y) = \frac{\sum_y \exp\left(f_\theta(\mathbf{x})[y]\right)}{Z(\theta)}$$




Your Classifier is Secretly an Energy Based Model and You Should Treat it Like One, ICLR 2020

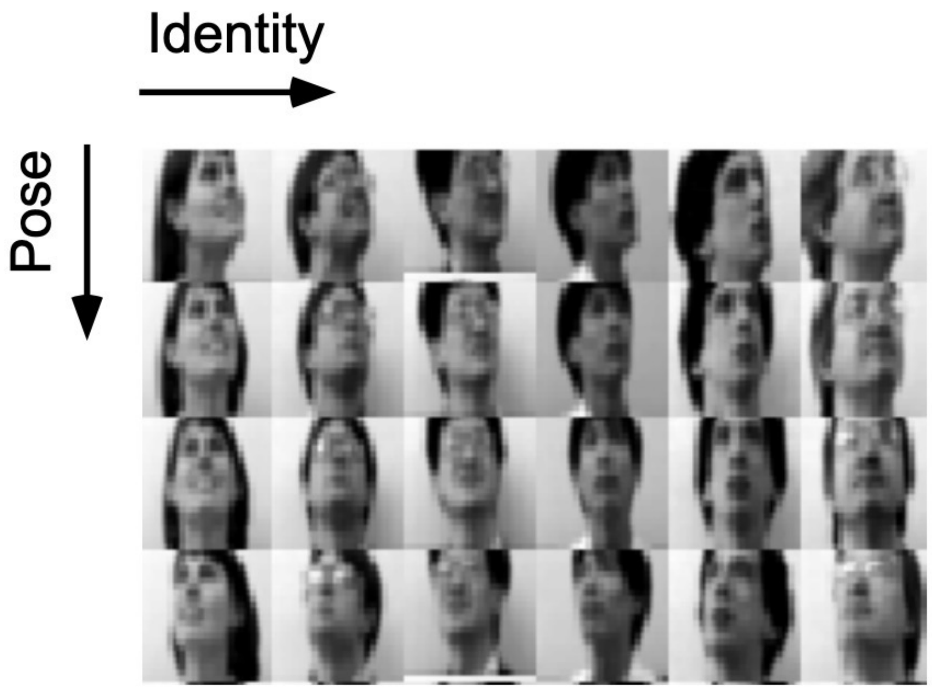
https://proceedings.neurips.cc/paper_files/paper/2023/file/b87bdcf963cad3d0b265fcb78ae7d11e-Paper-Conference.pdf

## Probabilistic modeling

- Where does probability come from?

- Assuming underlying distributions of data generation process
  - *latent factors z (pose, lighting, scale, ...)*
  - *z has simple distributions*
  - *observations x are rendered by a "world model" that's a function on z*
  - *observations x have complex distributions*

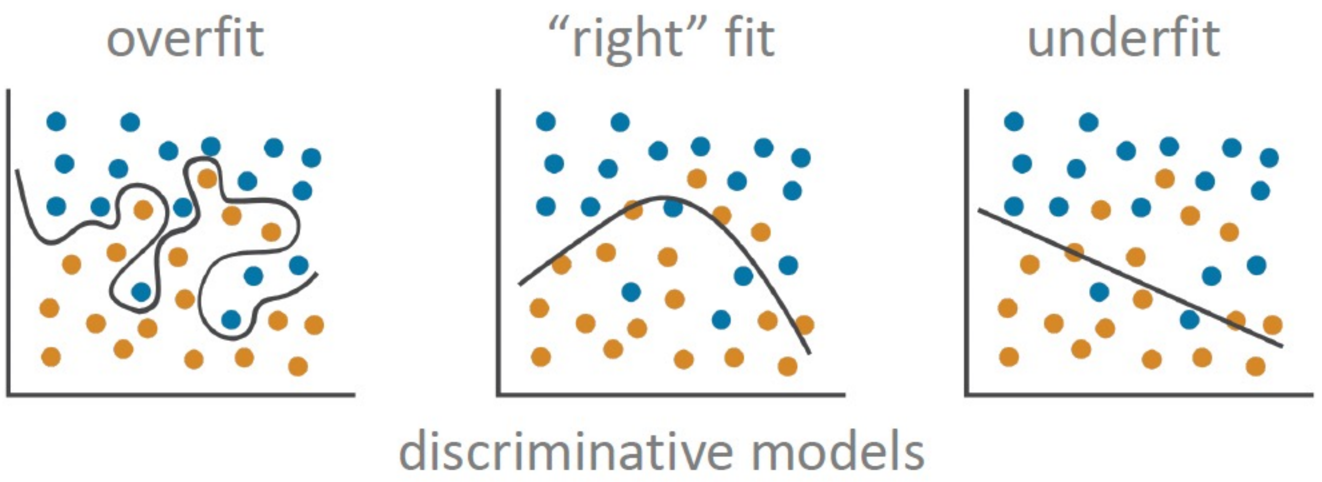
- Probability is part of the modeling.




https://www.merl.com/publications/docs/TR96-37.pdf

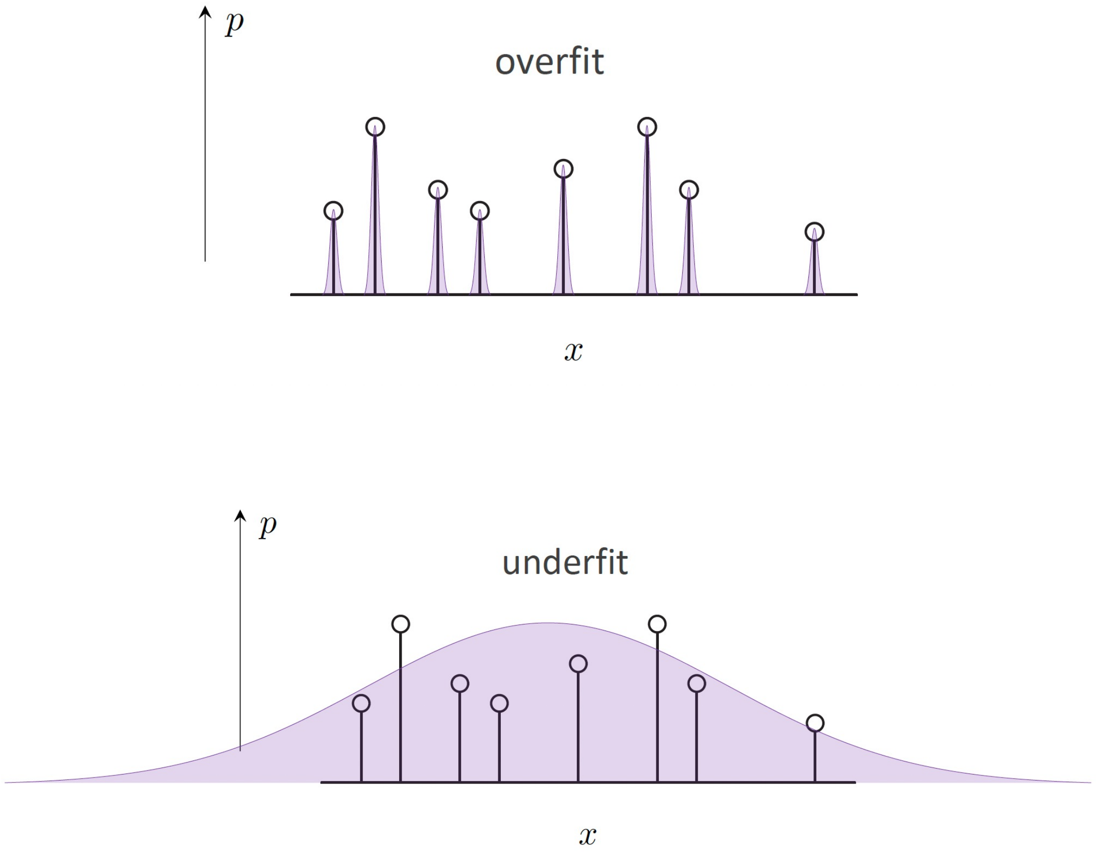
## Probabilistic modeling

- There may not be "underlying" distributions
- Even there are, what we can observe are a finite set of data points
- The models extrapolate the observations for modeling distributions
- Overfitting vs. underfitting: like discriminative models

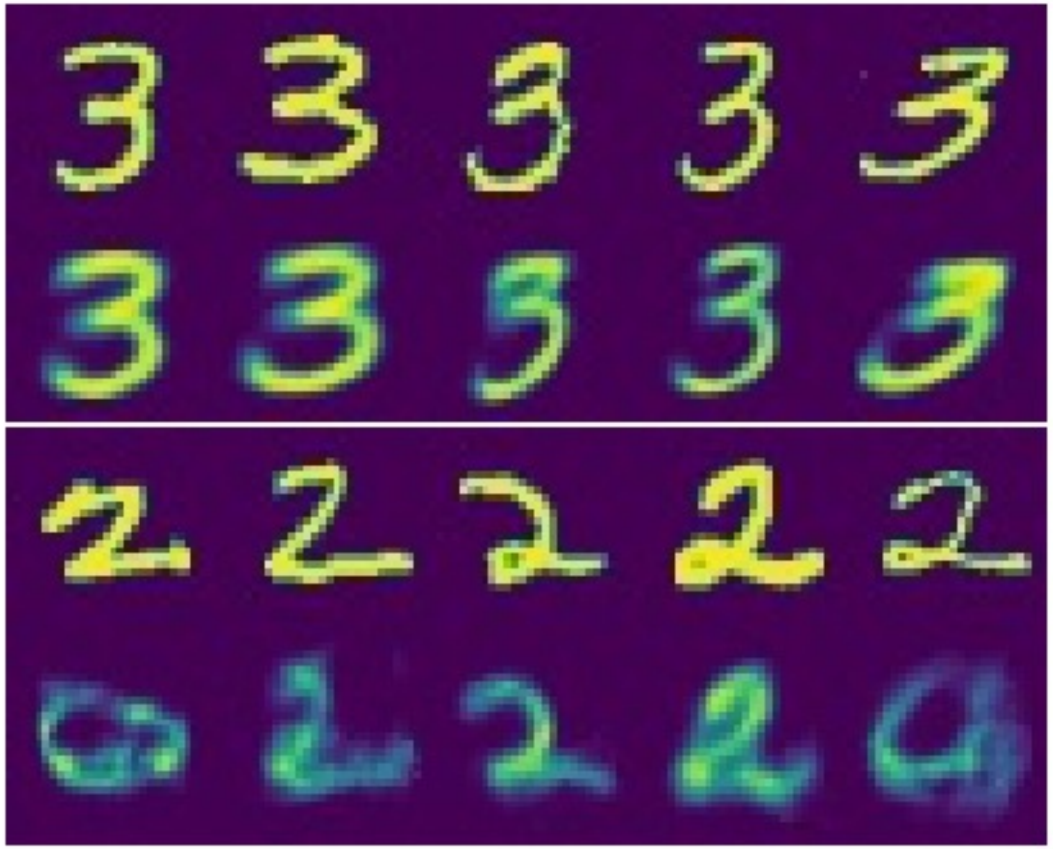



https://www.mathworks.com/discovery/overfitting.html

## Probabilistic modeling




https://arxiv.org/abs/2004.05675



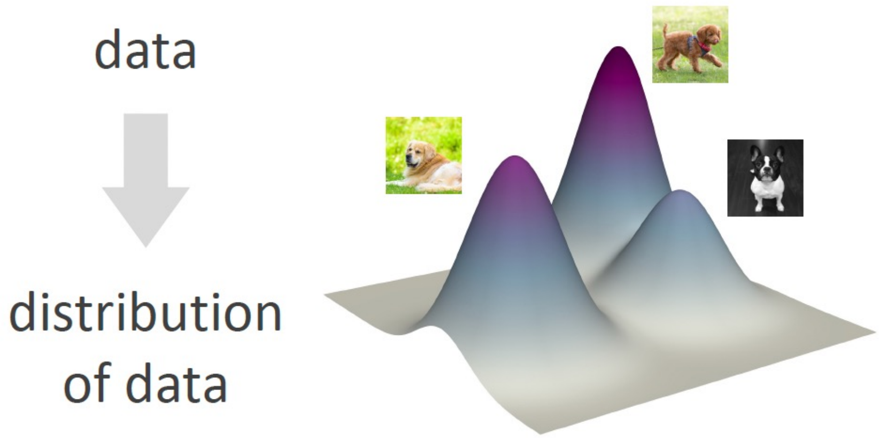
A Non-Parametric Test to Detect Data-Copying in Generative Models, AISTATS 2020

# What are Generative Models?

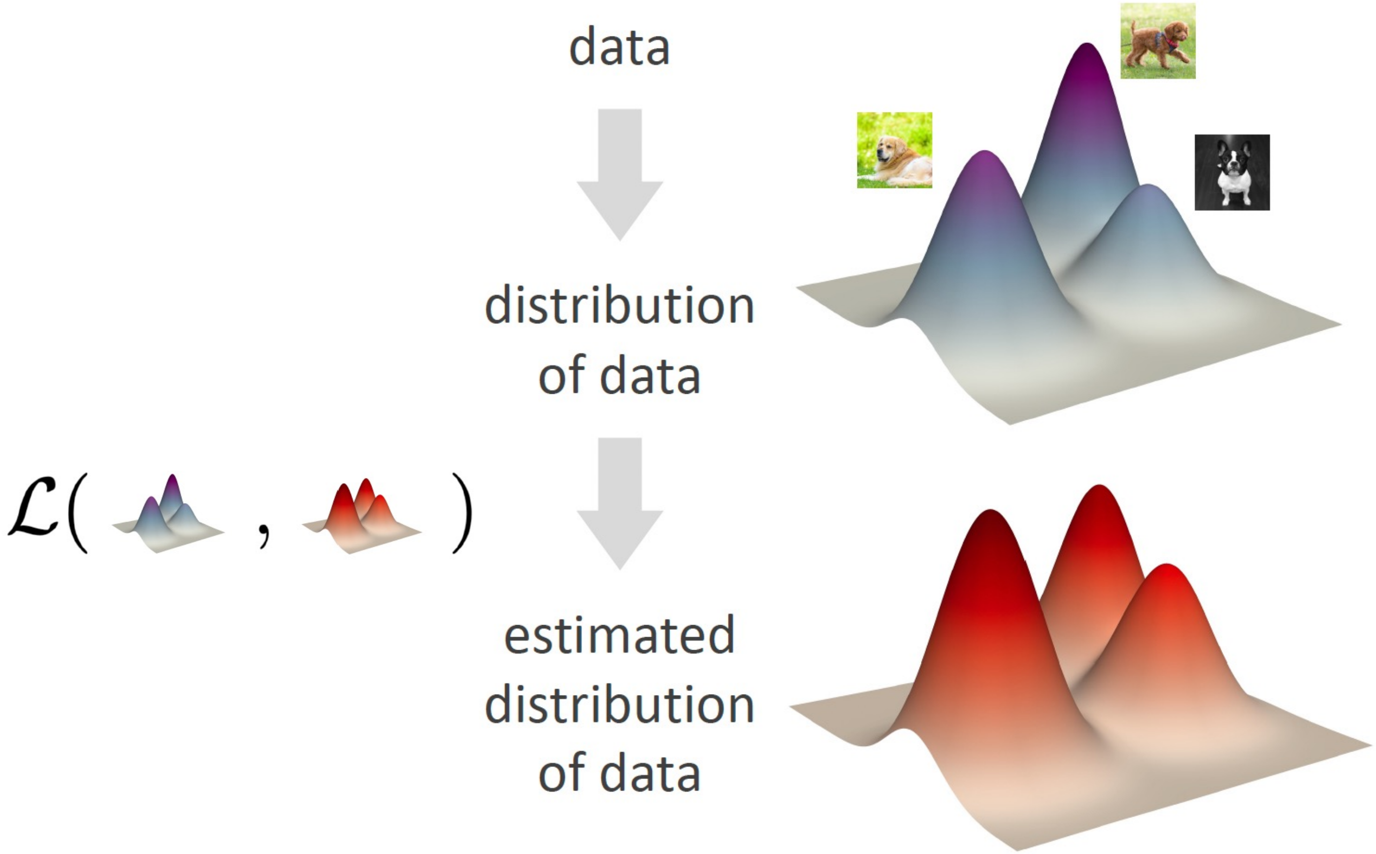
## Generative Models with Probabilistic modeling

data

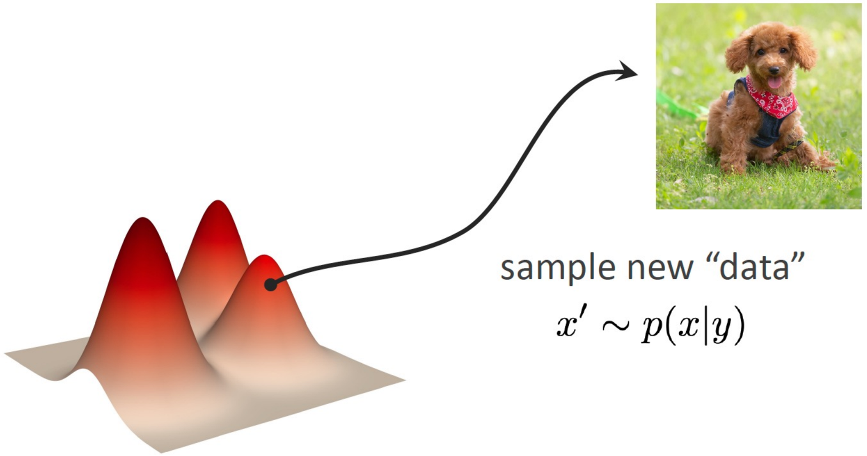
## Generative Models with Probabilistic modeling

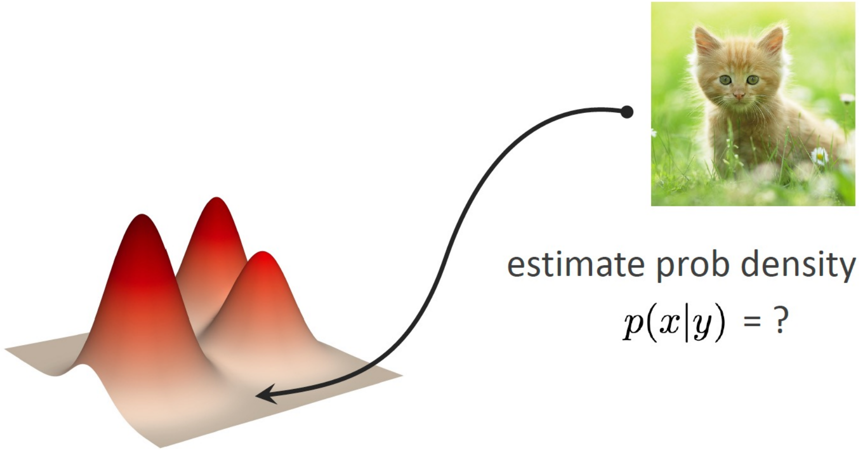


## Generative Models with Probabilistic modeling



## Generative Models with Probabilistic modeling



## Generative Models with Probabilistic modeling

- Generative models involve statistical models which are often designed and derived by humans
- Probabilistic modeling is not just the work of neural nets

## Generative Models with Probabilistic modeling

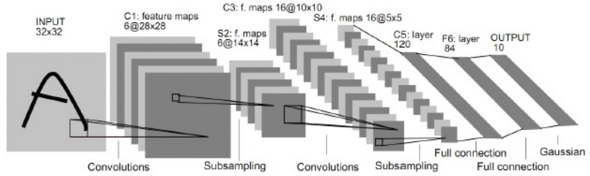
- Generative models involve statistical models which are often designed and derived by humans
- Probabilistic modeling is not just the work of neural nets
- Probabilistic modeling is a popular way, but not the only way
- "*All models are wrong, but some are useful.”* - George Box

# What are Deep Generative Models?

# Deep Generative Models

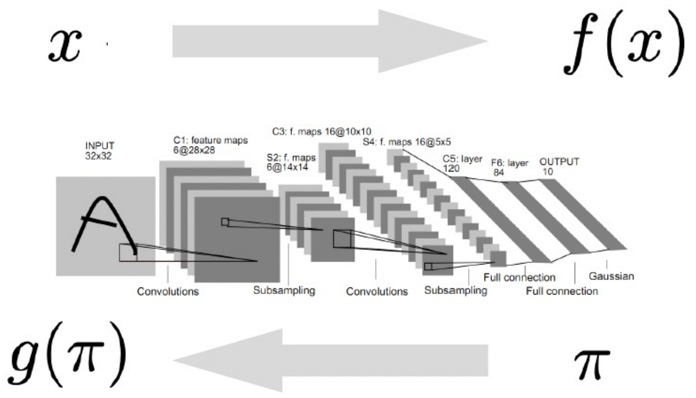
- Deep learning is representation learning
- Learning to represent data instances
  - map data to feature: $x \rightarrow f(x)$
  - minimize loss w/ target: $\mathcal{L}(y, f(x))$

$x \longrightarrow f(x)$



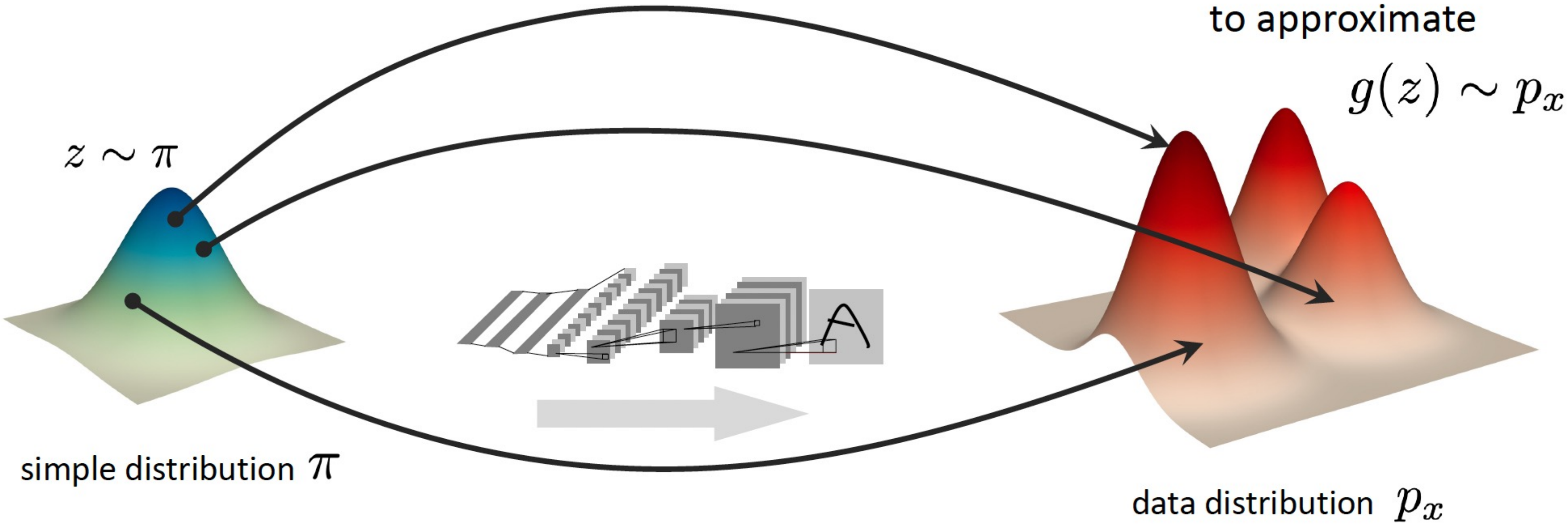
## Deep Generative Models

- Deep learning is representation learning
- Learning to represent data instances
  - map data to feature: $x \to f(x)$
  - minimize loss w/ target: $\mathcal{L}(y, f(x))$
- Learning to represent probability distributions
  - map a simple distribution (Gaussian/uniform) to a complex one: $\pi \to g(\pi)$
  - minimize loss w/ data distribution: $\mathcal{L}(p_x, g(\pi))$
- Often perform both together



## Deep Generative Models

- Learning to represent probability distributions
  - From simple to complex distributions

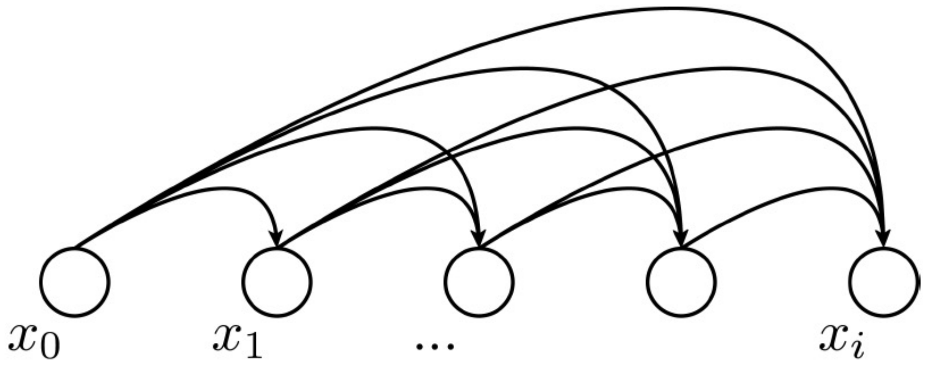


## Deep Generative Models

- Learning to represent probability distributions
  - From simple to complex distributions
  - Not all parts of distribution modeling is done by learning!

**Case study:**
**Autoregressive model**

This dependency graph is designed (not learned).

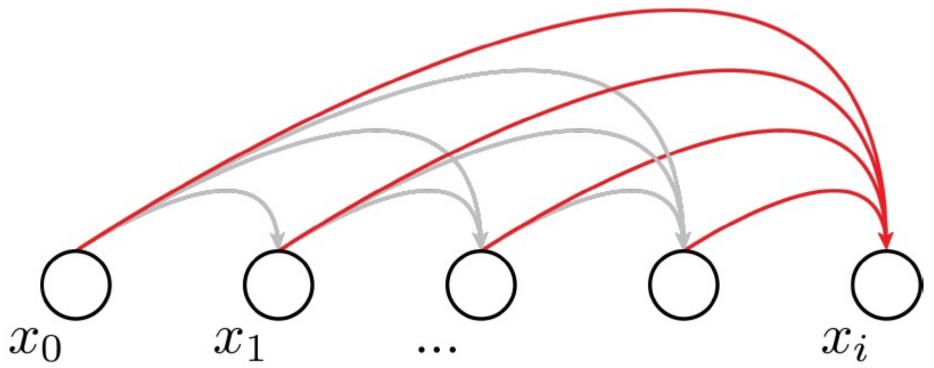


## Deep Generative Models

- Learning to represent probability distributions
    - From simple to complex distributions
    - Not all parts of distribution modeling is done by learning!

**Case study:**
**Autoregressive model**

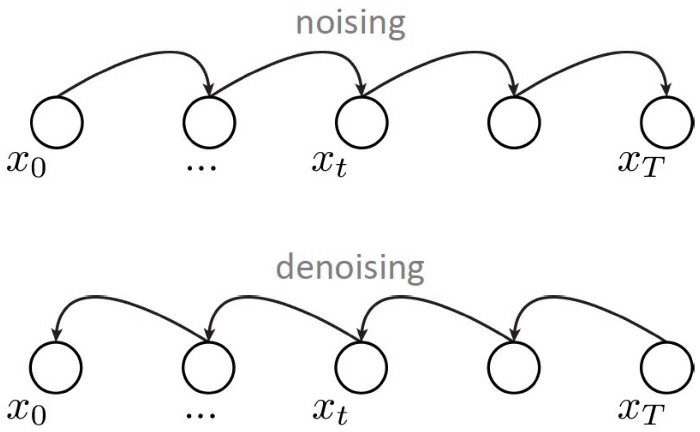
The mapping function is learned (e.g., Transformer)



## Deep Generative Models

- Learning to represent probability distributions
  - From simple to complex distributions
  - Not all parts of distribution modeling is done by learning!

**Case study:**
**Diffusion model**

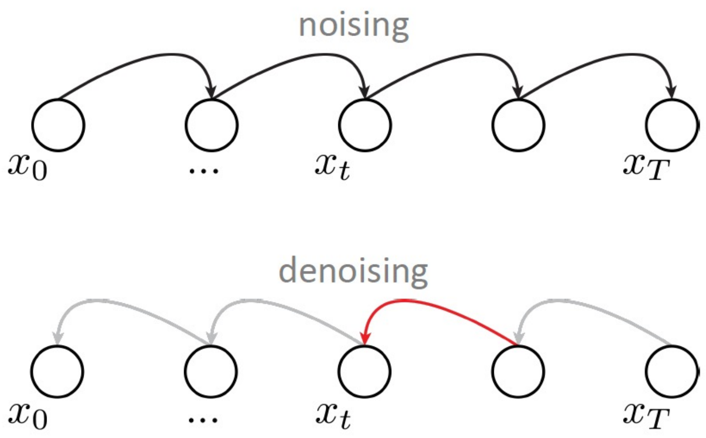



This dependency graph is designed (not learned).

## Deep Generative Models

- Learning to represent probability distributions
  - From simple to complex distributions
  - Not all parts of distribution modeling is done by learning!

**Case study:**
**Diffusion model**




The mapping function is learned (e.g., Unet)

## Summary: Deep Generative Models may involve:

- **Formulation:**
  - formulate a problem as probabilistic modeling
  - decompose complex distributions into simple and tractable ones
- **Representation:**
  - deep neural networks to represent data and their distributions
- **Objective function:**
  - to measure how good the predicted distribution is
- **Optimization:**
  - optimize the networks and/or the decomposition
- **Inference:**
  - sampler: to produce new samples
  - probability density estimator (optional)

## Formulating Real-world Problems as Generative Models

# Real-world Examples

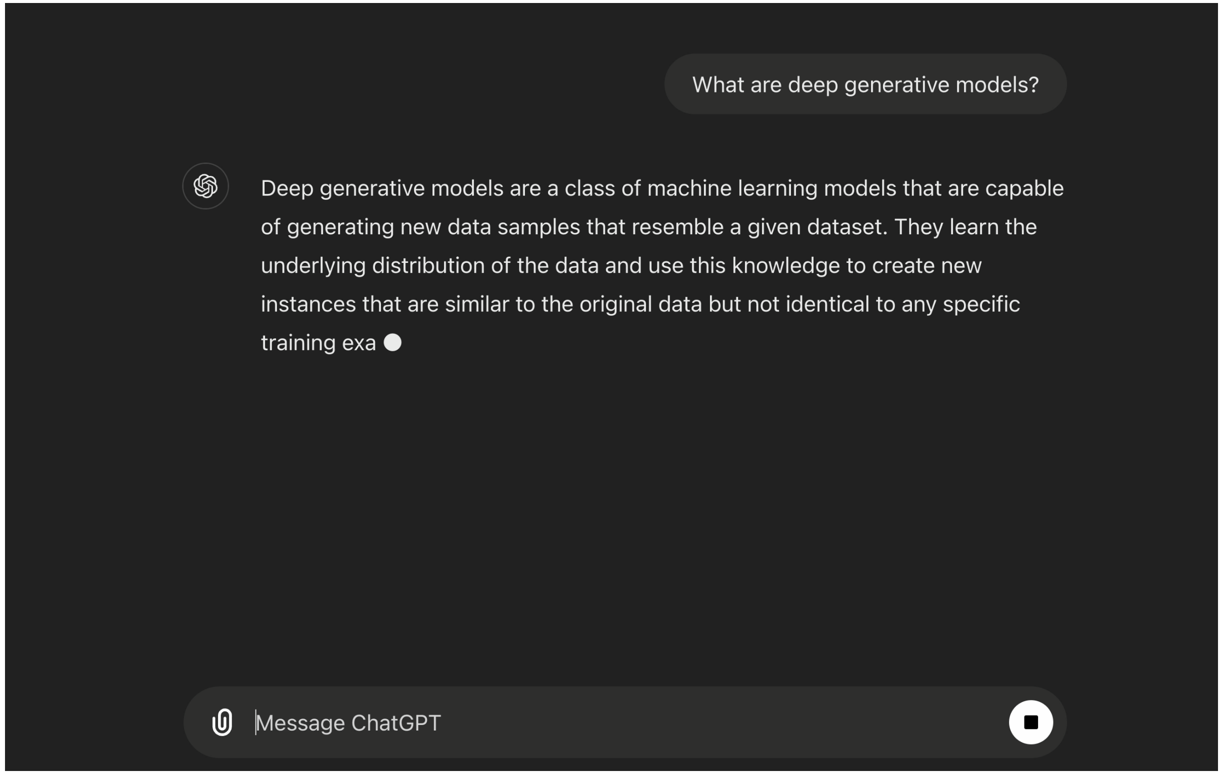
- Generative models are about $p(x|y)$

What can be y?
- Condition
- Constraint
- Label
- Partial Observation
- …
- Less informative

What can be x?
- Data
- Sample
- Observation
- Full Observation
- …
- More informative

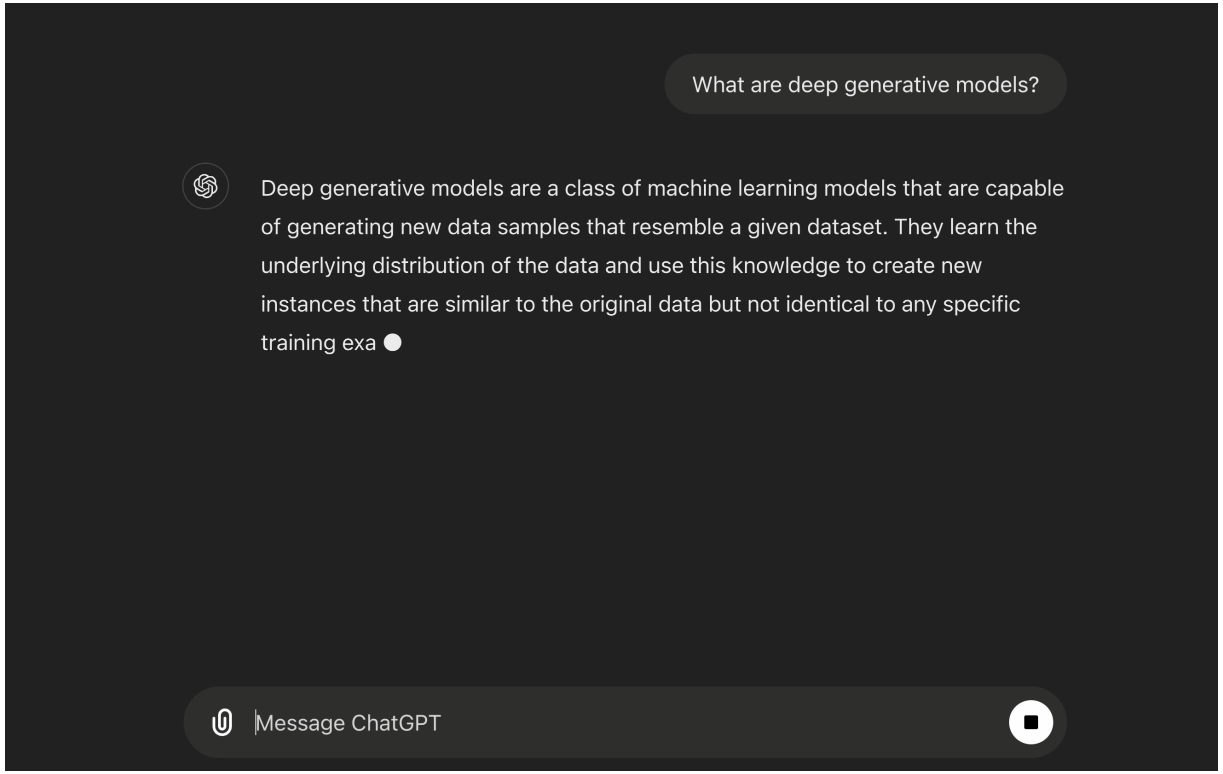
# Real-world Examples

- Natural language conversation
  - y: prompt
  - x: response of the chatbot



# Real-world Examples

- Natural language conversation
  - y: prompt
  - x: response of the chatbot
  - Really?



# Real-world Examples

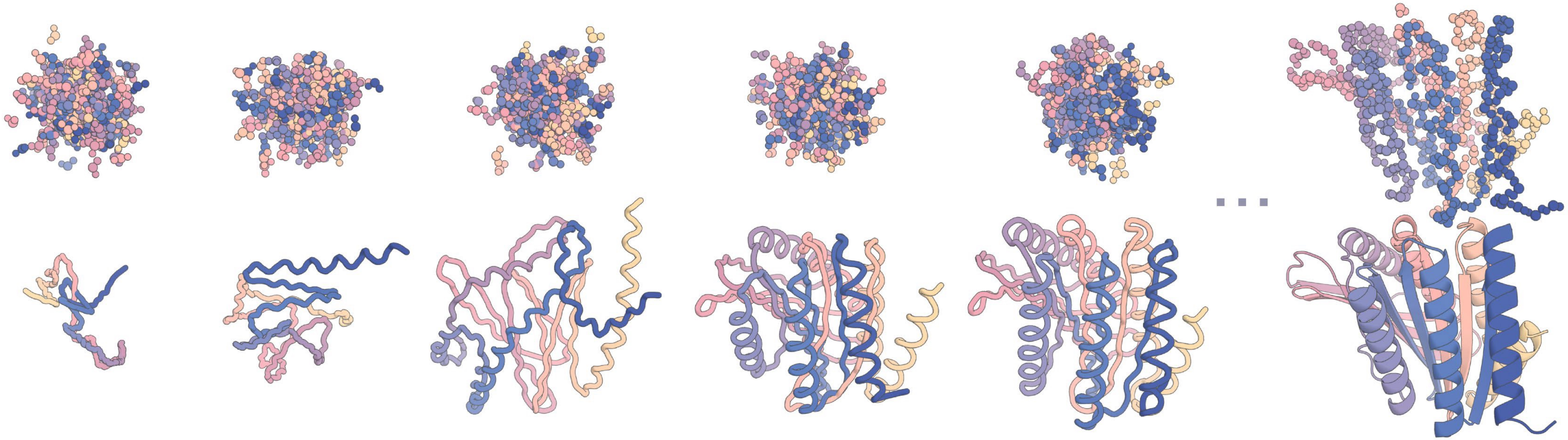
- Text-to-image/video generation
  - y: text prompt, e.g. *a panda is teaching a course named "deep generative models"*
  - x: generated image/video


# Real-world Examples

- Text-to-image/video generation
  - y: text prompt, e.g. *a panda is teaching a course named "deep generative models"*
  - x: generated image/video
  - Really?


# Real-world Examples

- Protein structure generation
  - y: condition/constraint, e.g. *symmetry*
  - x: generated protein structures

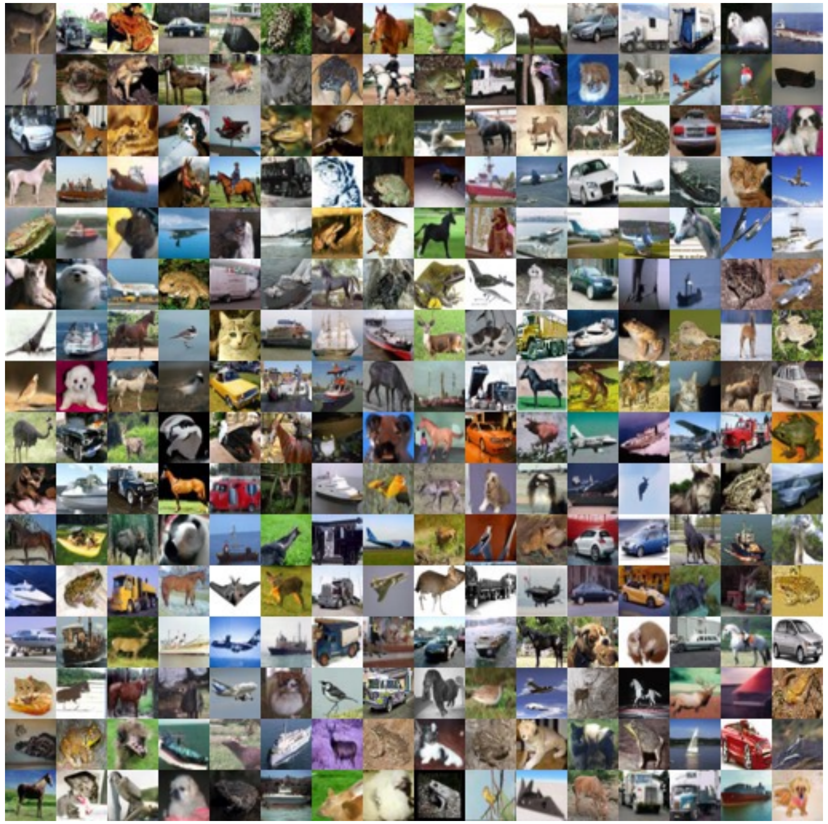


# Real-world Examples

- Class-conditioned image generation
  - y: class label, e.g. *dog*
  - x: generated image

# Real-world Examples

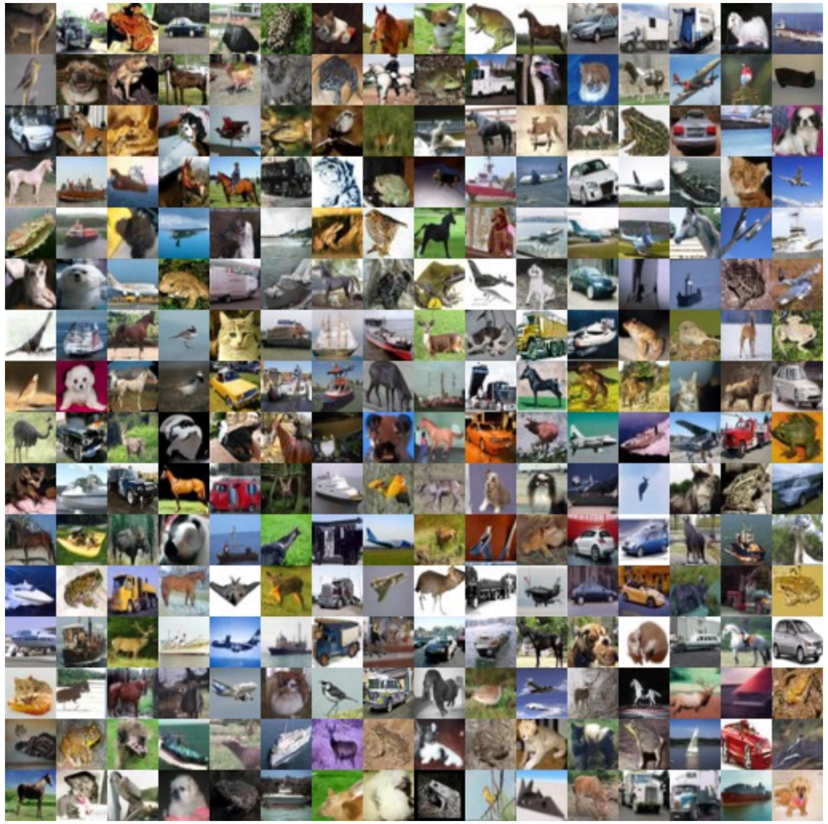
- Unconditional image generation
  - y: ?
  - x: generated image



https://arxiv.org/abs/2206.00364

# Real-world Examples

- Unconditional image generation
  - y: y = which dataset
  - x: generated image



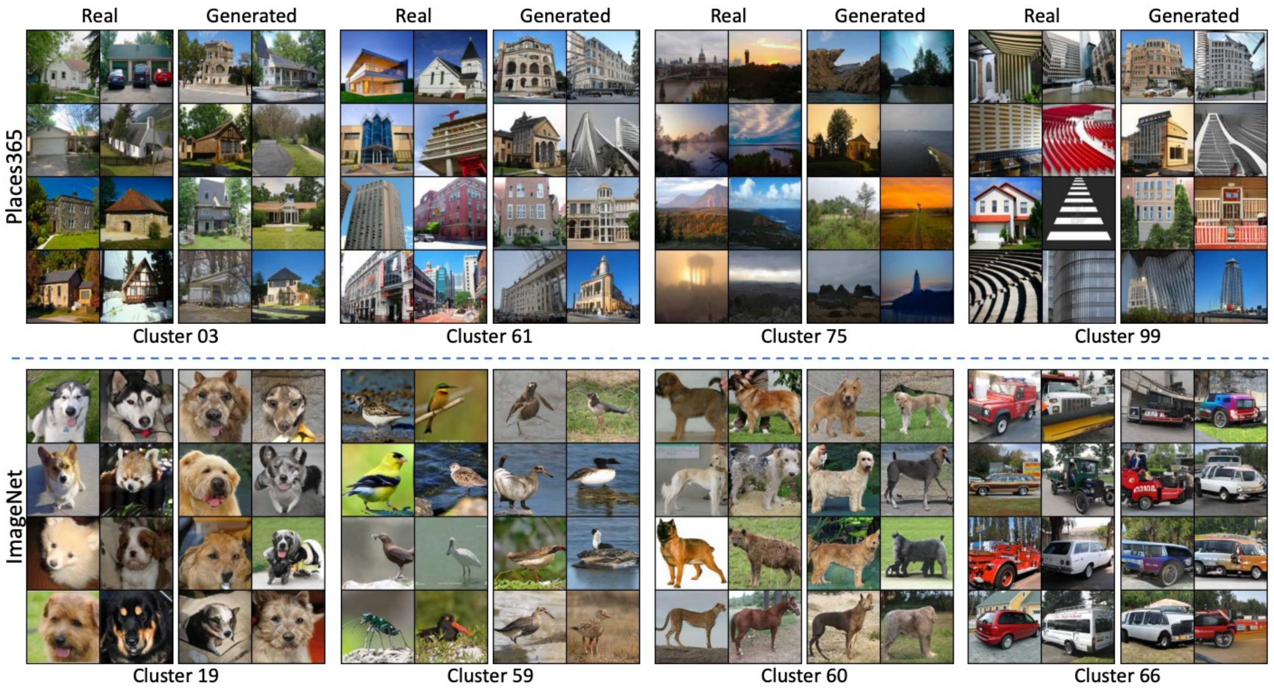
$p(x|y)$: images ~ CIFAR10

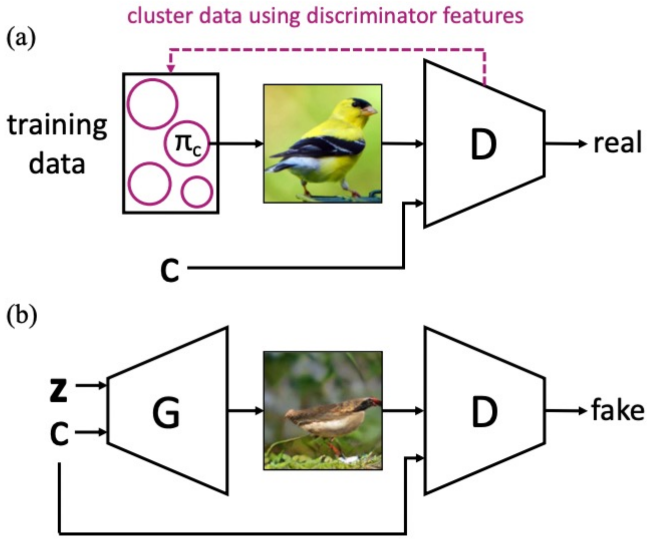
$p(x)$: all images

https://arxiv.org/abs/2206.00364

# Real-world Examples

- Unconditional image generation
  - y: some clusters
  - x: generated image




Diverse Image Generation via Self-Conditioned GANs, CVPR 2020

# Real-world Examples

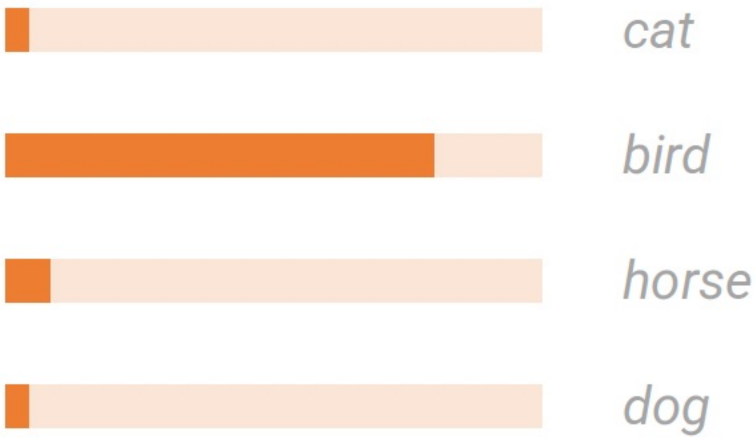
- Unconditional image generation
  - y: each training image
  - x: generated image

$$\min_{\theta \in \Theta} \frac{1}{N} \sum_{i=1}^{N} \left[ \min_{z_i \in \mathcal{Z}} \ell\left(g_\theta(z_i), x_i\right) \right]$$

Optimizing the Latent Space of Generative Networks, ICML 2018

# Real-world Examples

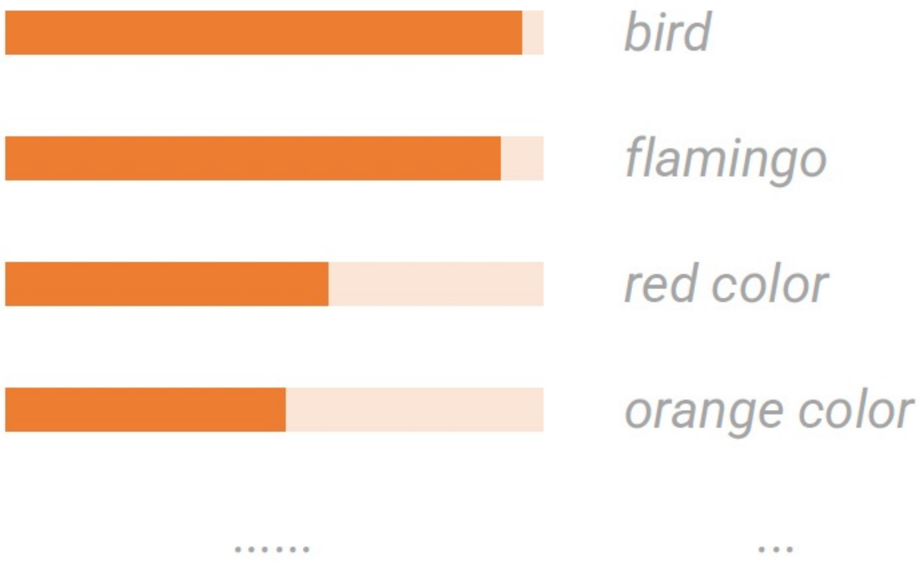
- Image classification
  - y: input image
  - x: generated class labels



HKU Musketeers Foundation Institute of Data Science 香港大學同心基金數據科學研究院

# Real-world Examples

- Open-vocabulary recognition
  - y: input image
  - x: generated class labels
  - From fixed labels to open ones



# Real-world Examples

- Image captioning
  - y: input image
  - x: generated descriptions
  - From open labels to free-form descriptions

```
a baseball player with a catcher and umpire on top of a baseball field.
a baseball player is sliding into a base.
a baseball player swings at a pitch with the pitcher and umpire behind him.
baseball player with bat in the baseball game.
a batter in the process on the bat in a baseball game.
```

# Real-world Examples

- Multimodal model
  - y: input image and some query
  - x: generated responses
  - From descriptions to any responses

**GPT-4 visual input example, Extreme Ironing:**

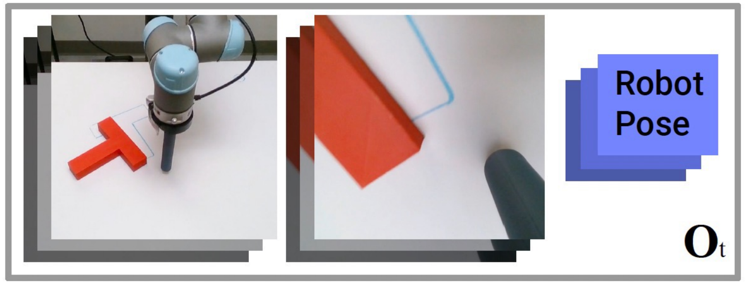
User What is unusual about this image?

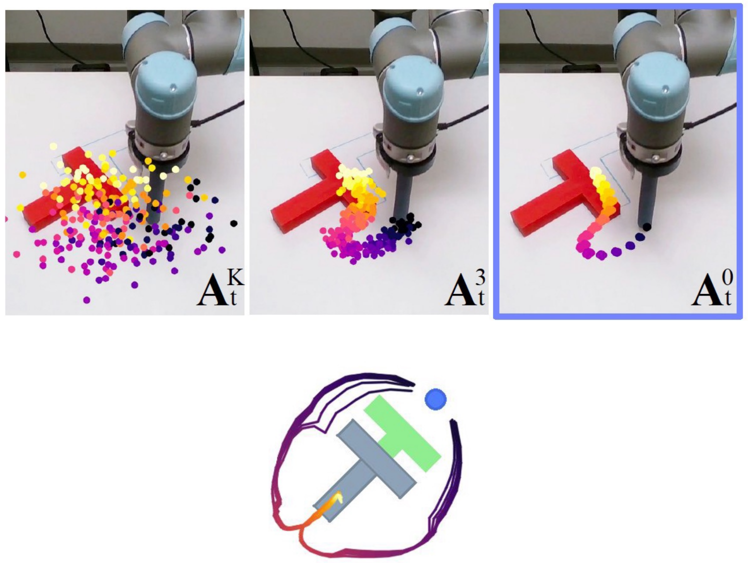
Source: https://www.barnorama.com/wp-content/uploads/2016/12/03-Confusing-Pictures.jpg

GPT-4: The unusual thing about this image is that a man is ironing clothes on an ironing board attached to the roof of a moving taxi.

# Real-world Examples

- Policy learning in Robotics
  - y: visual and other sensory observations
  - x: policies (probability of actions)



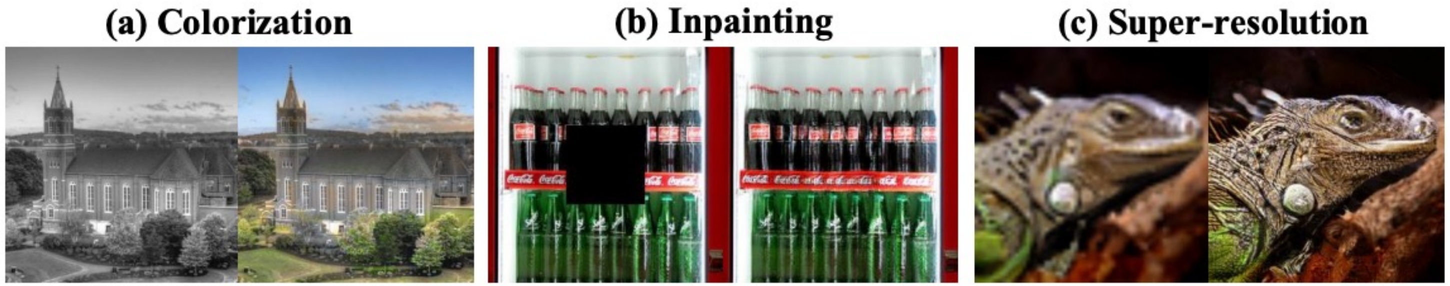

Diffusion Policy

Diffusion Policy: Visuomotor Policy Learning via Action Diffusion, RSS 2023

HKU Musketeers Foundation Institute of Data Science 香港大學同心基金數據科學研究院

# Real-world Examples

- Image restoration
  - y: broken image
  - x: restored image

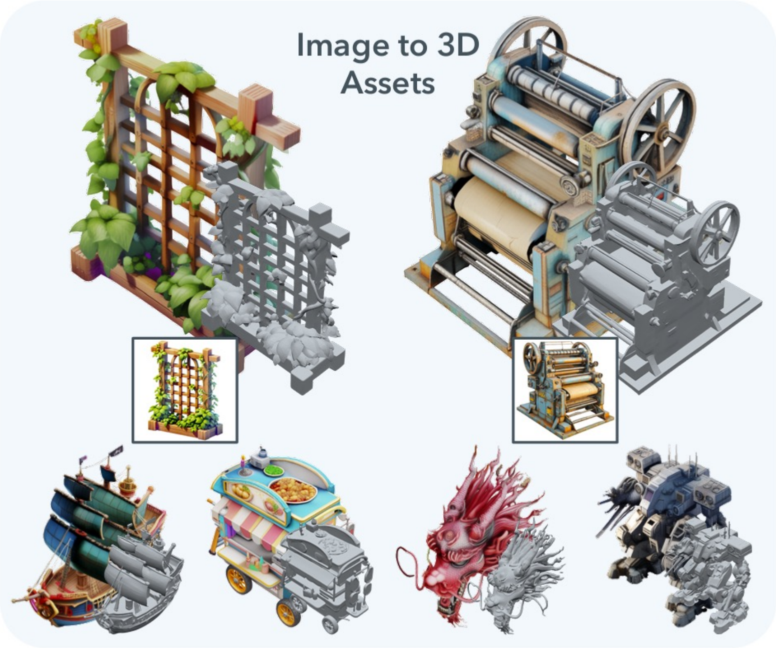



$$\mathbf{z}^* = \arg\min_{\mathbf{z}\in\mathbb{R}^d} E(\hat{\mathbf{x}}, G(\mathbf{z};\boldsymbol{\theta})), \qquad \mathbf{x}^* = G(\mathbf{z}^*;\boldsymbol{\theta}),$$
$$= \arg\min_{\mathbf{z}\in\mathbb{R}^d} \mathcal{L}(\hat{\mathbf{x}}, \phi(G(\mathbf{z};\boldsymbol{\theta}))),$$

Exploiting Deep Generative Prior for Versatile Image Restoration and Manipulation, ECCV 2020

# Real-world Examples

- Dimension extension
    - y: low-dimensional observation
    - x: actual high-dimensional data




Structured 3D Latents for Scalable and Versatile 3D Generation, CVPR 2025

# Real-world Examples

- Many problems can be formulated as generative models. Key research questions:
  - What is x? What is y?
  - How to represent x, y, and their dependence?