

https://introml.mit.edu

6.390 Intro to Machine Learning

Lecture 8: Representation Learning

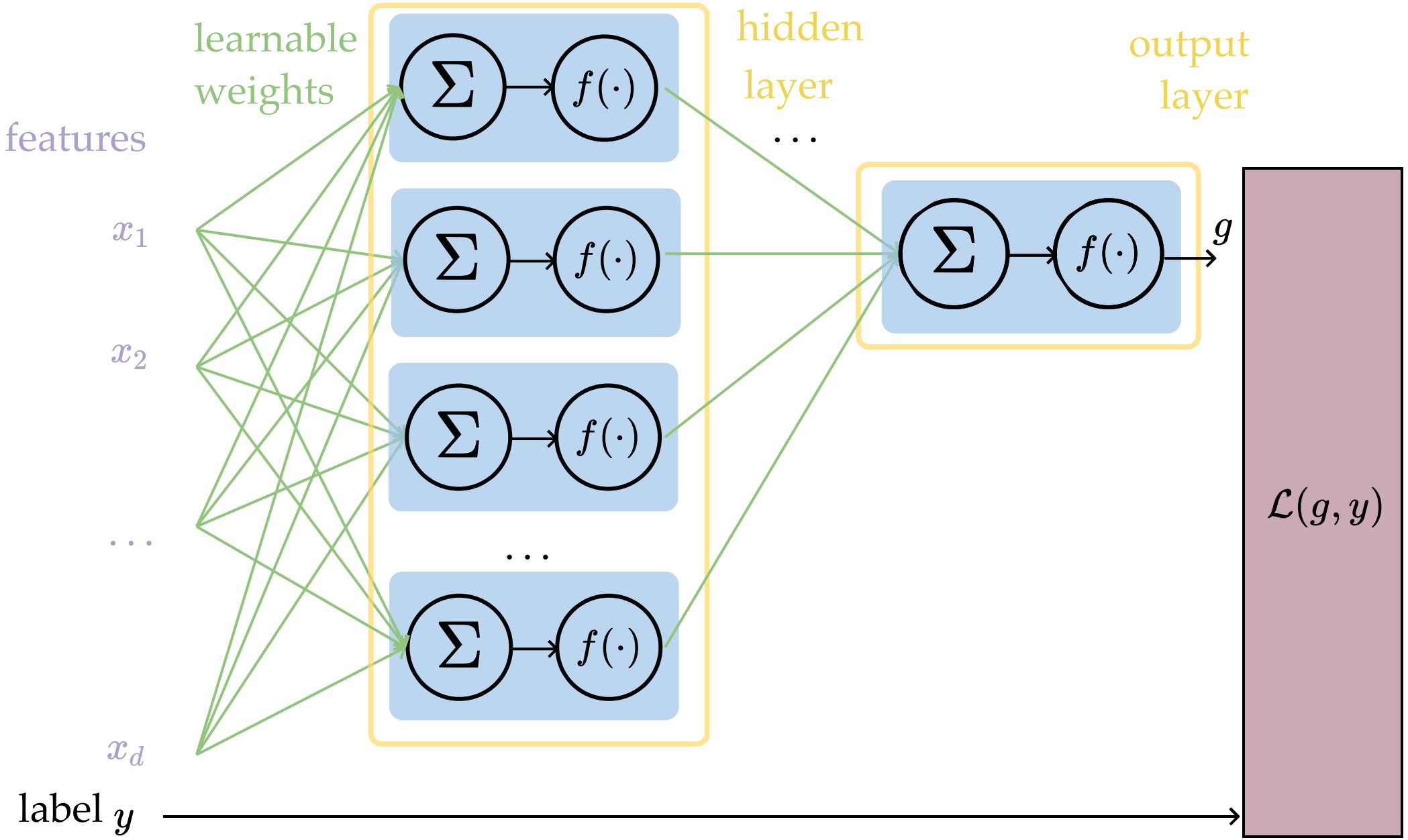
Shen Shen

Oct 26, 2026

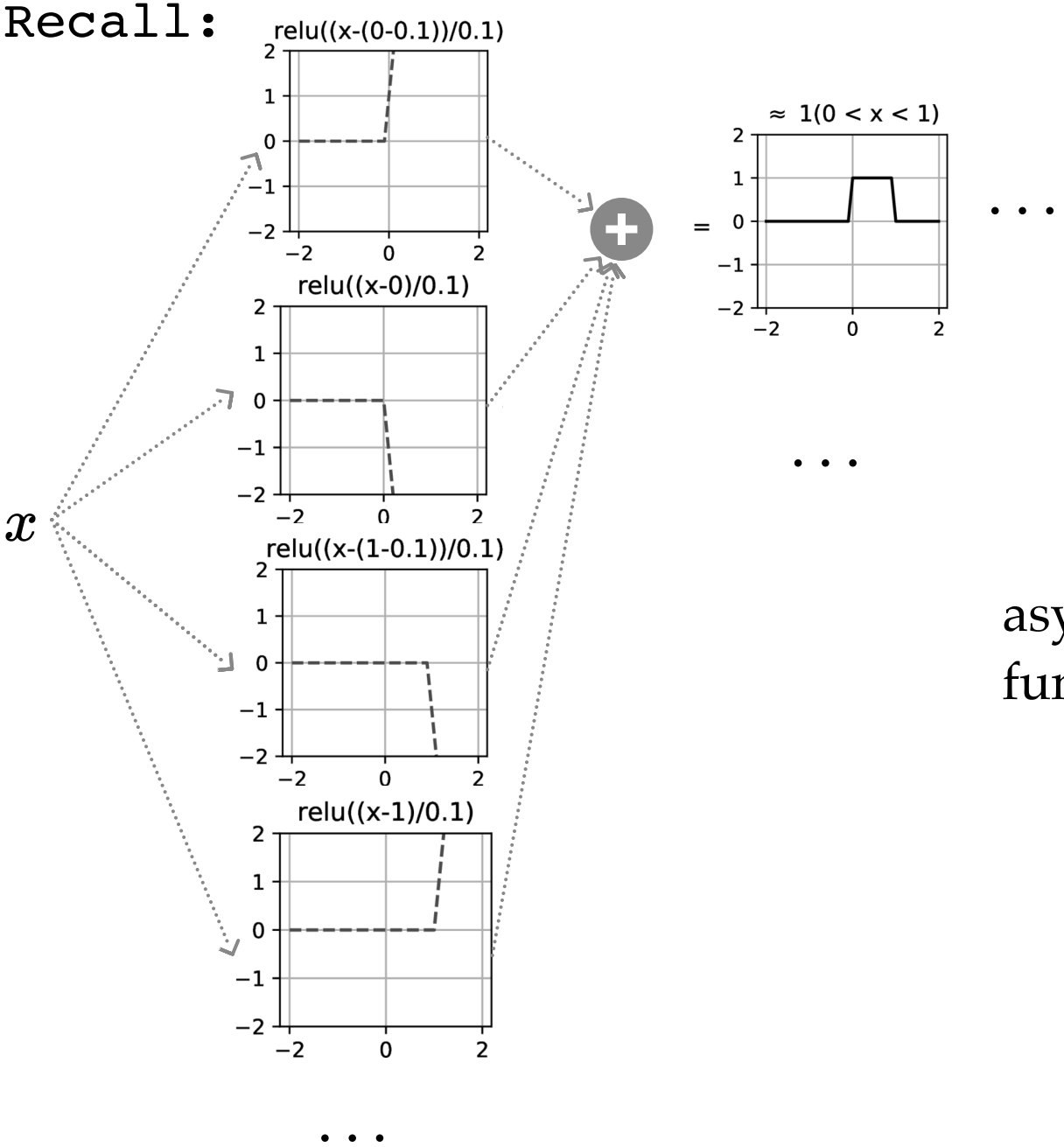
2:30pm, Room 10-250

Slides and Lecture Recording

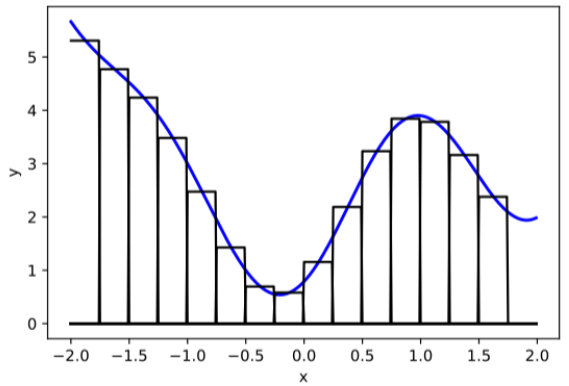
Recall:


learnable
weights
hidden
layer
output
layer
features
x_1
x_2
$\cdots$
x_d
Σ
$f(\cdot)$
g
$\mathcal{L}(g, y)$
label y

Recall:



asymptotically, can approximate any smooth function arbitrarily well!



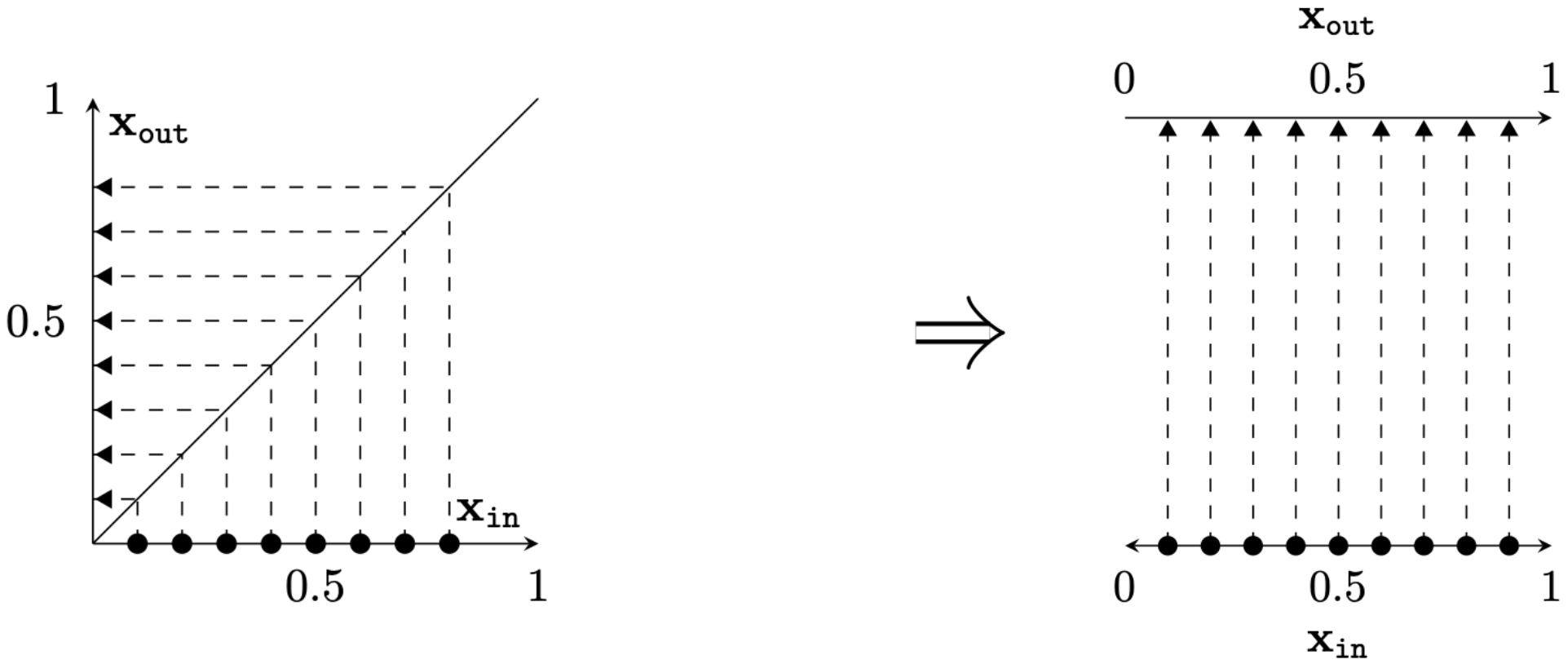
Outline

- Neural networks are *representation* learners
- Auto-encoders and self-supervised learning
- Word embeddings
- Multi-modality

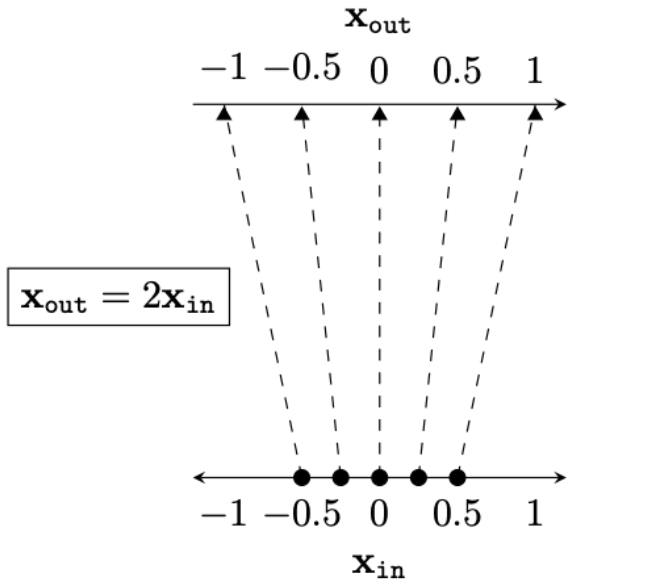
[some images in parts 1–2; credit: visionbook.mit.edu]

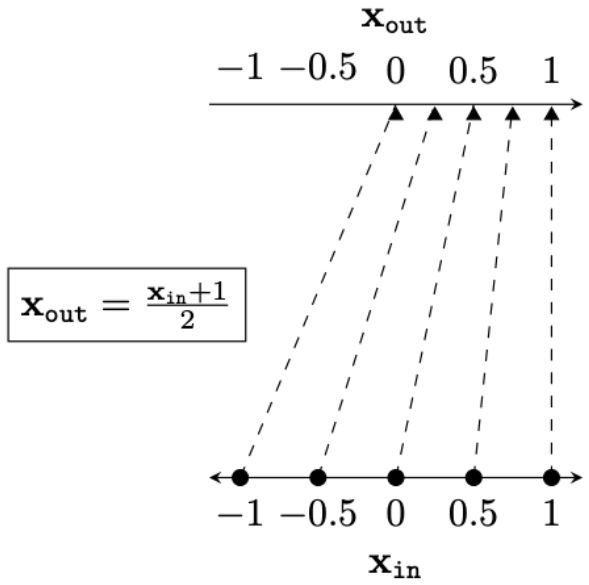
Two different ways to visualize a function

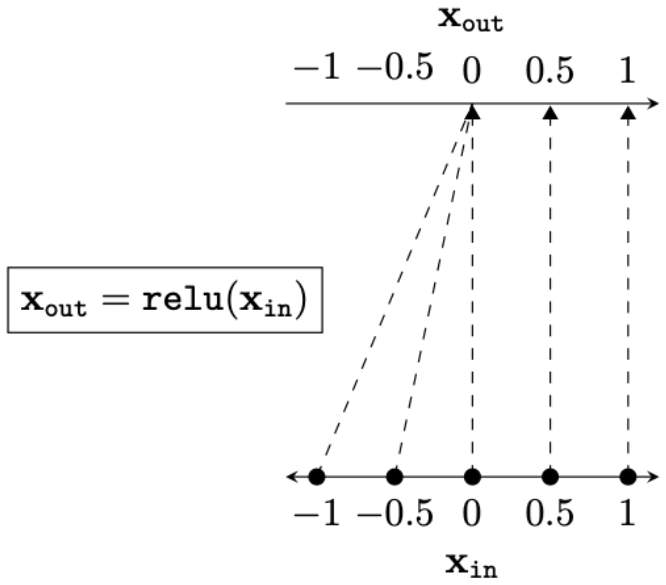
e.g. the identity function

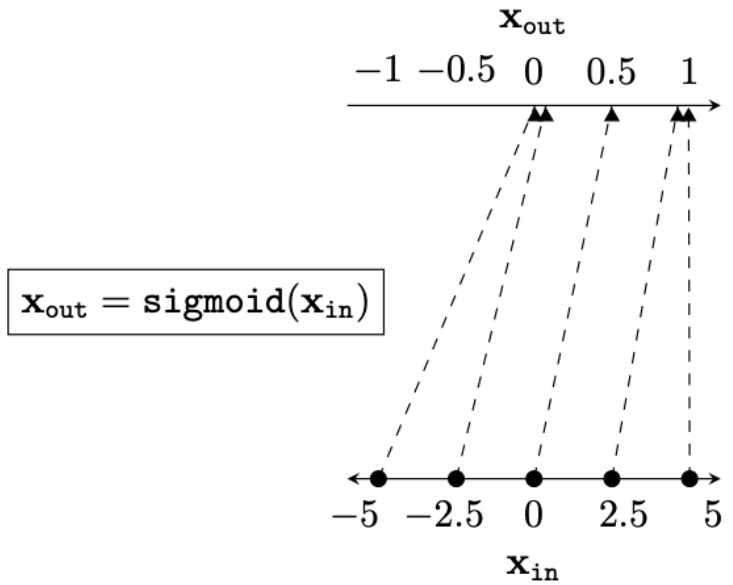


Representation transformations for a variety of neural net operations

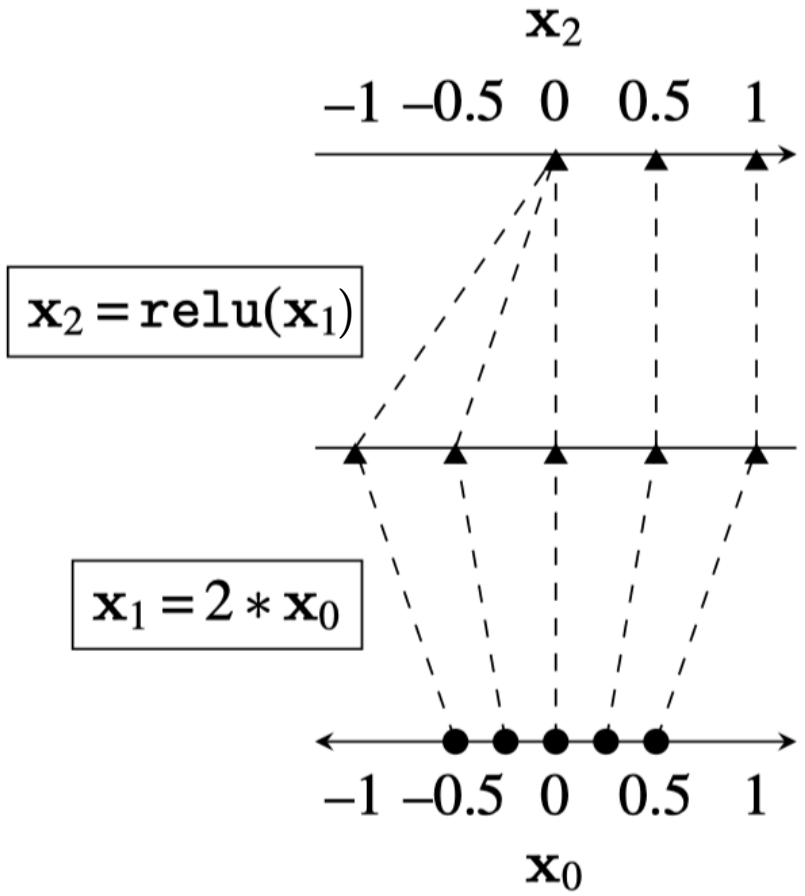




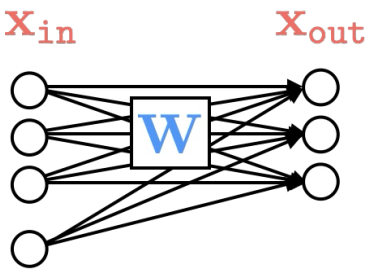
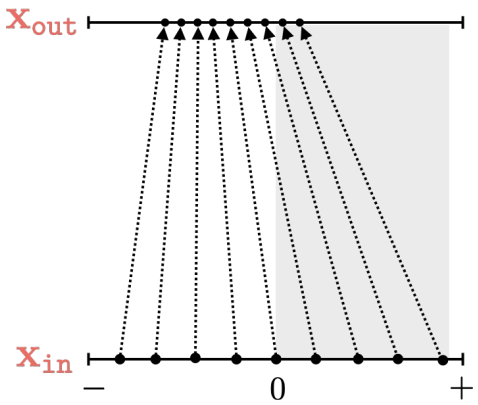
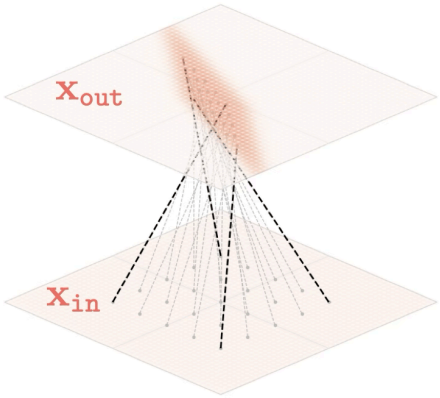
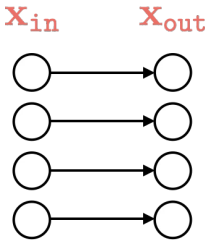
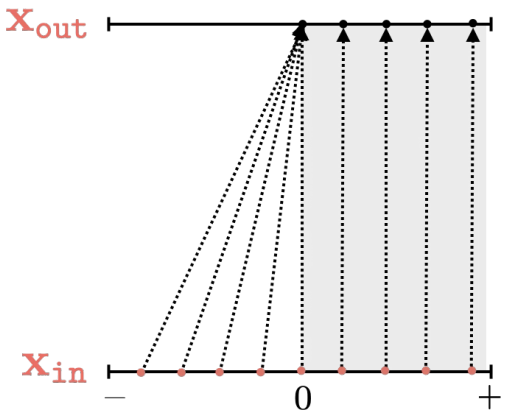
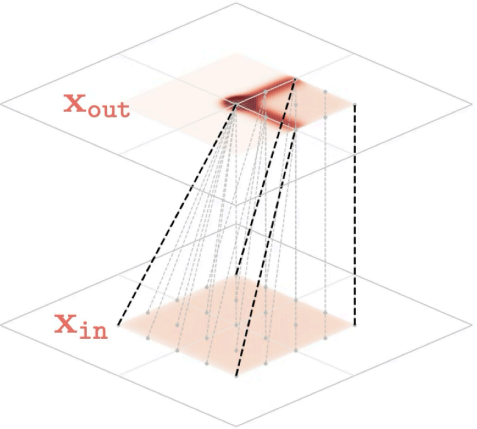




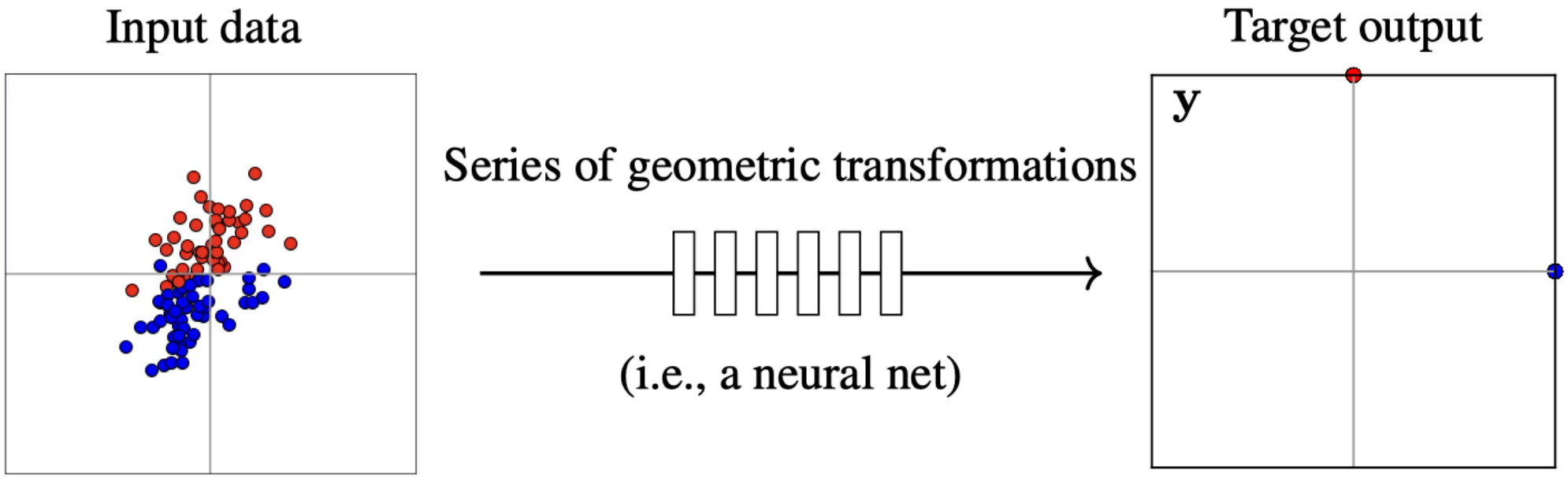
and a stack of neural net operations



Parameters

	wiring graph	equation	mapping 1D	mapping 2D
linear		$\mathbf{x}_{out} = \mathbf{W}\mathbf{x}_{in}$		
relu		$x_{out_i} = \max(x_{in_i}, 0)$		

What does training a neural net classifier look like?



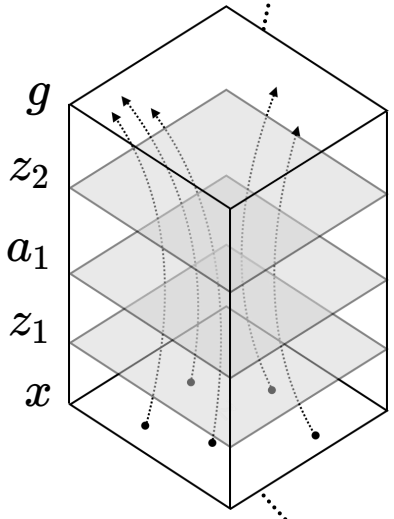
$$g = \text{softmax}(z_2)$$

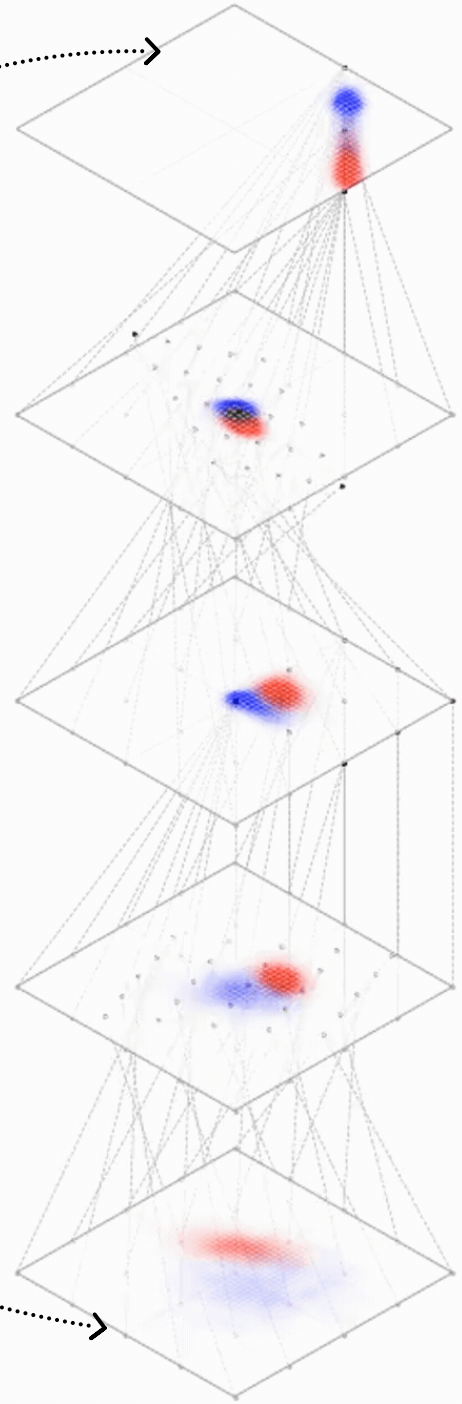
$$z_2 = \text{linear}\,(a_1) \in \mathbb{R}^2$$

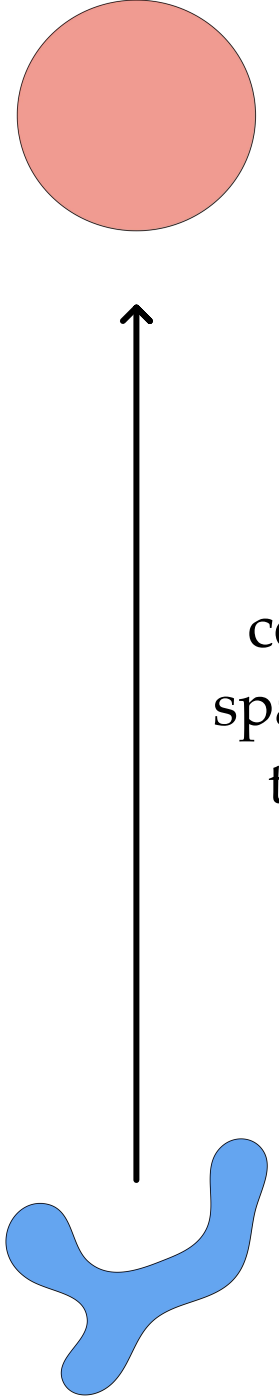
$$a_1 = \text{ReLU}(z_1)$$

$$z_1 = \text{linear}\,(x) \in \mathbb{R}^2$$

$$x \in \mathbb{R}^2$$

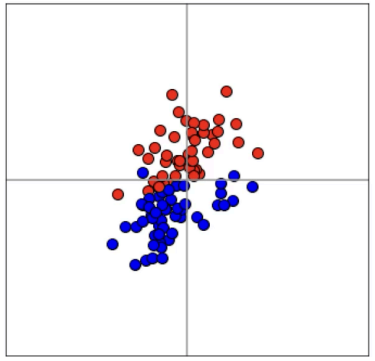


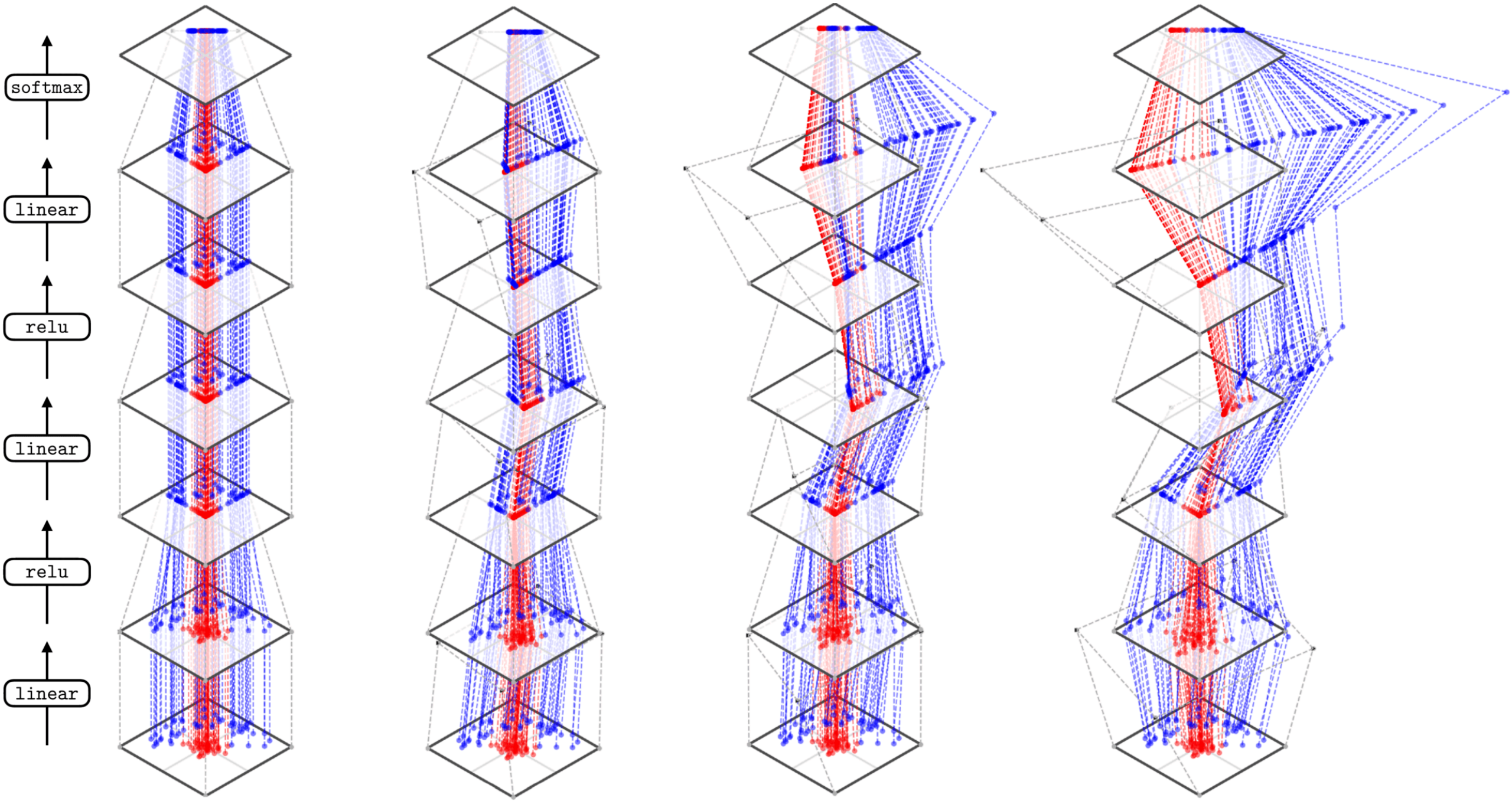




maps from complex data space to desired target space

Training data



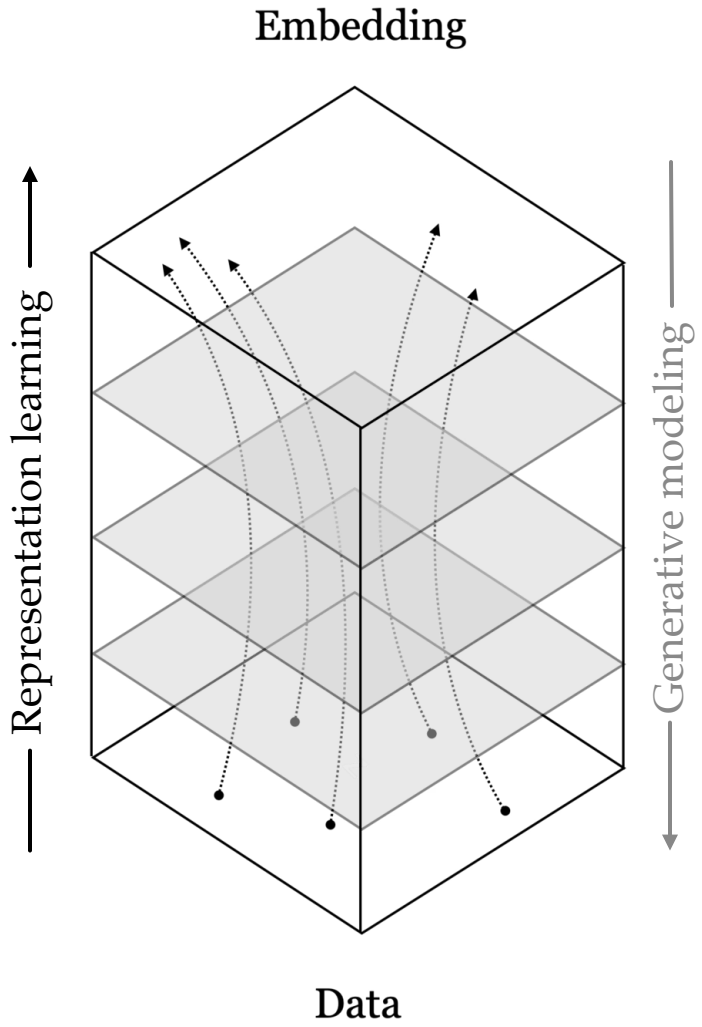

softmax
linear
relu
linear
relu
linear
Training iteration

Deep neural nets transform datapoints, layer by layer

Each layer is a different *representation,* aka *embedding,* of the data

From data to latent embeddings: representation learning

(From latent embeddings to data: generative modeling)



https://distill.pub/2017/feature-visualization/

ImageNet also taught us that labeling 14 million images by hand is brutal.

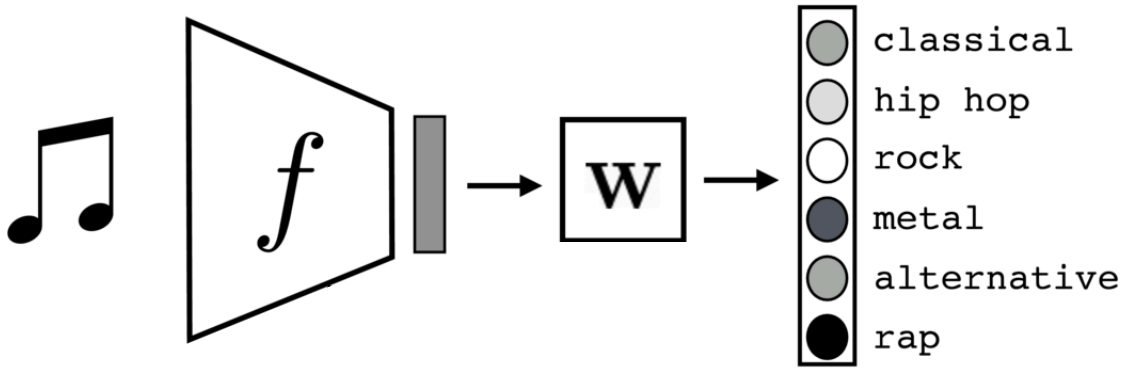
Undergrads were time-consuming, algorithms were flawed, and the team didn't have money—Li said the project failed to win any of the federal grants she applied for, receiving comments on proposals that it was shameful Princeton would research this topic, and that the only strength of proposal was that Li was a woman.

A solution finally surfaced in a chance hallway conversation with a graduate student who asked Li whether she had heard of Amazon Mechanical Turk, a service where hordes of humans sitting at computers around the world would complete small online tasks for pennies.

"He showed me the website, and I can tell you literally that day I knew the ImageNet project was going to happen," she said. "Suddenly we found a tool that could scale, that we could not possibly dream of by hiring Princeton undergrads."

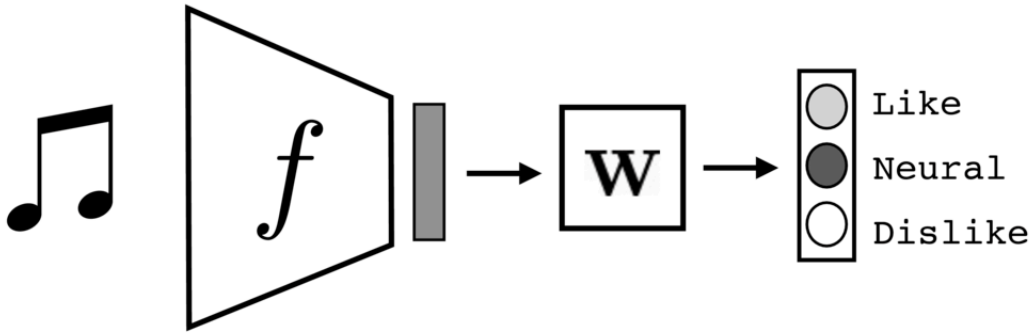
Training

Genre recognition



Testing

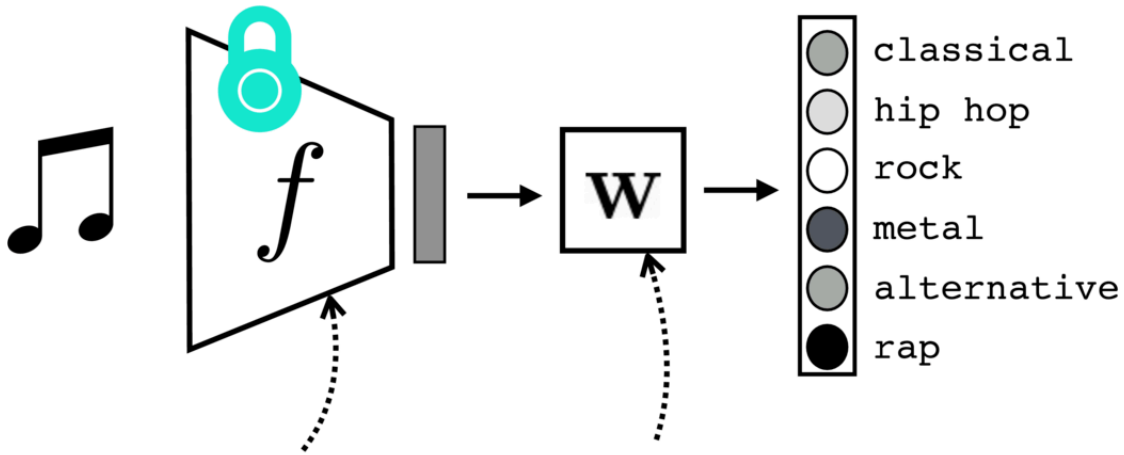
Preference prediction



Often, what we will be "tested" on is not what we were trained on.

Training

Genre recognition

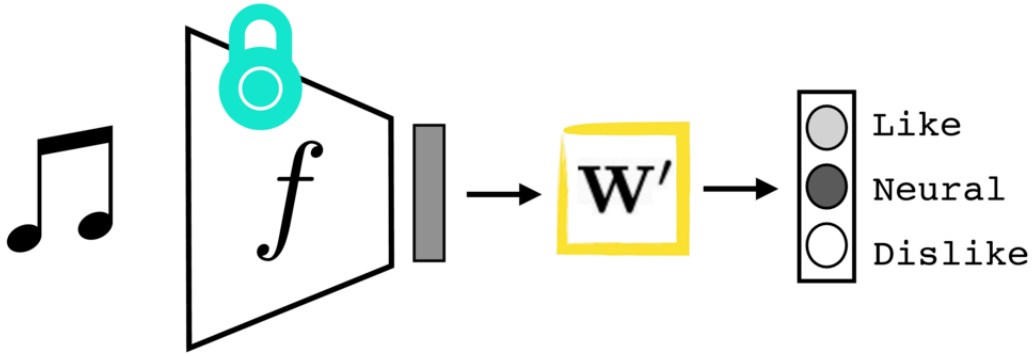


"common"-sense representation

task-specific prediction

Adapting

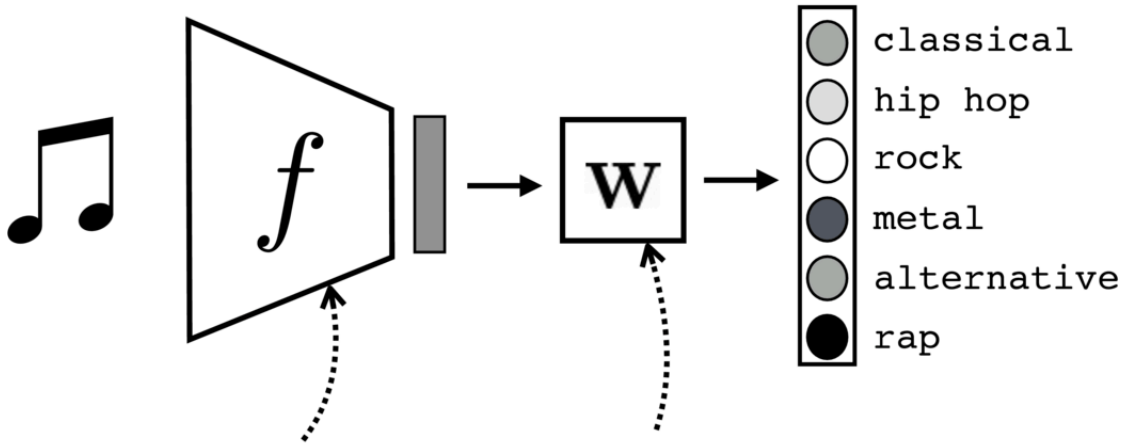
Preference prediction



Final-layer adaptation: freeze f, train a new final layer on new target data

Training

Genre recognition

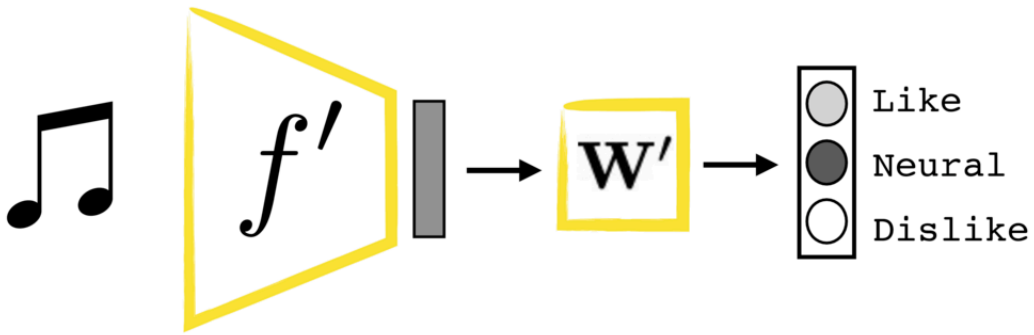


"common"-sense representation

task-specific prediction

Adapting

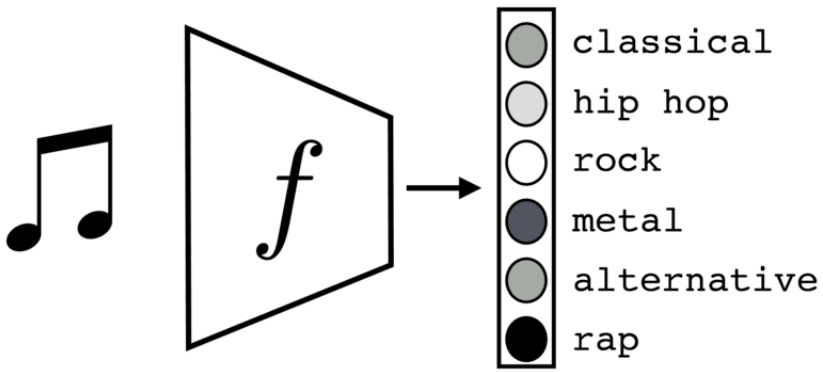
Preference prediction



Fine-tuning: initialize f' as f, then continue training for f' as well, on new target data

Pretraining

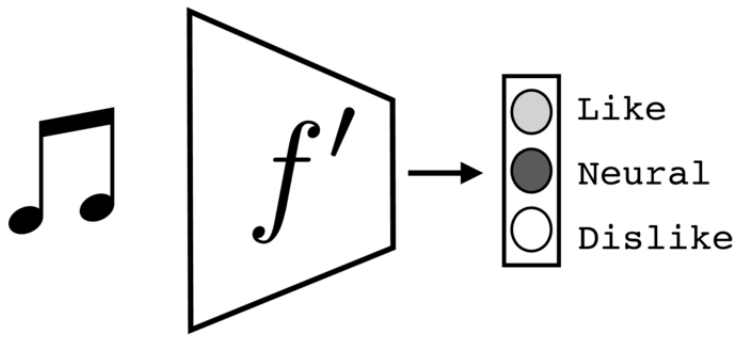
Genre recognition



A lot of data

Adapting

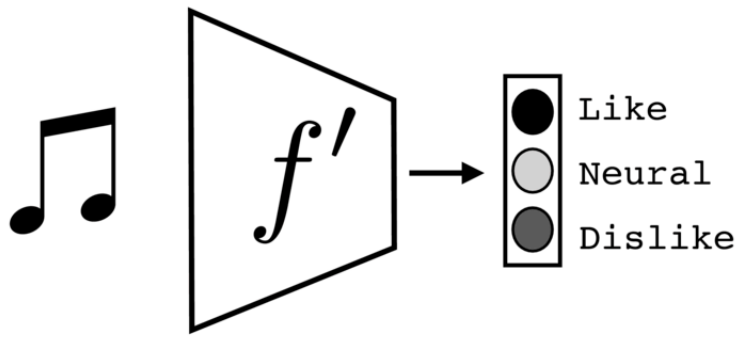
Preference prediction



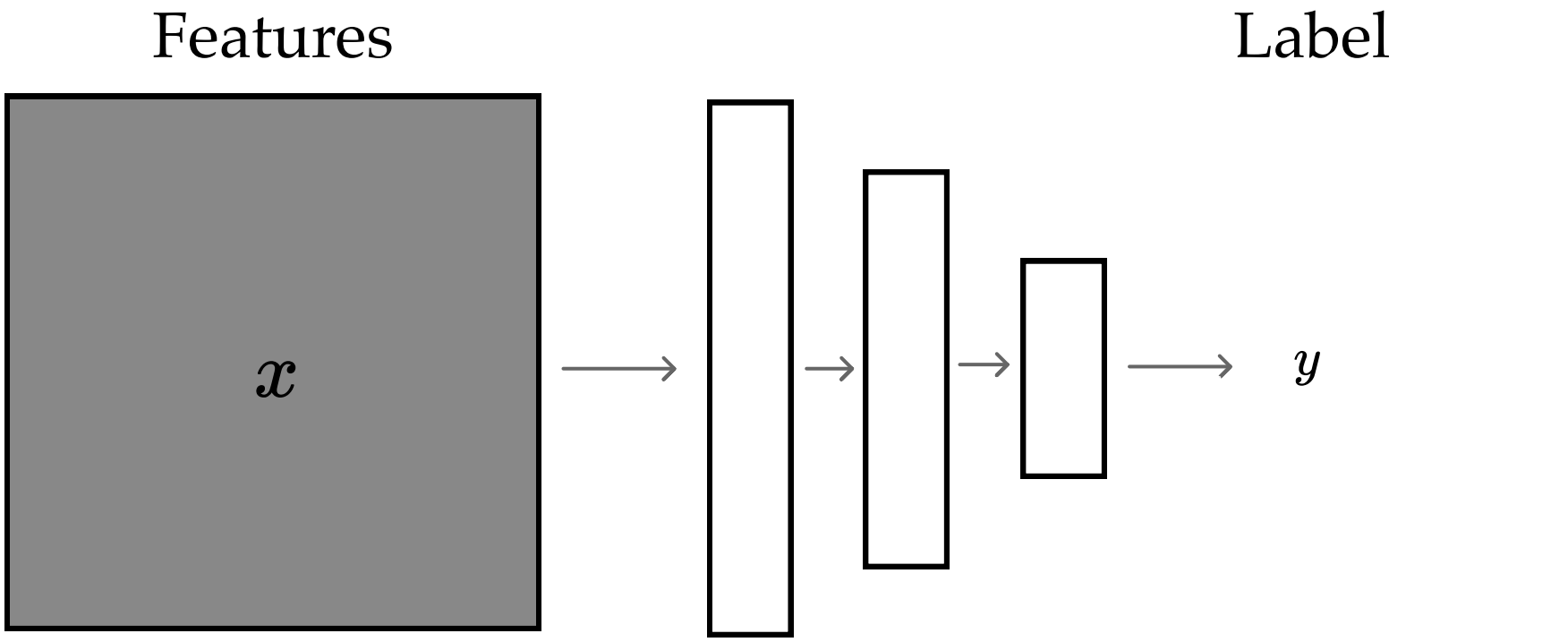
A little data

Testing

Preference prediction



Label prediction (supervised learning)



Unlabeled features x are abundant

Labels y are expensive…

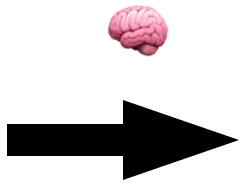
Can we learn useful representations without external labels?

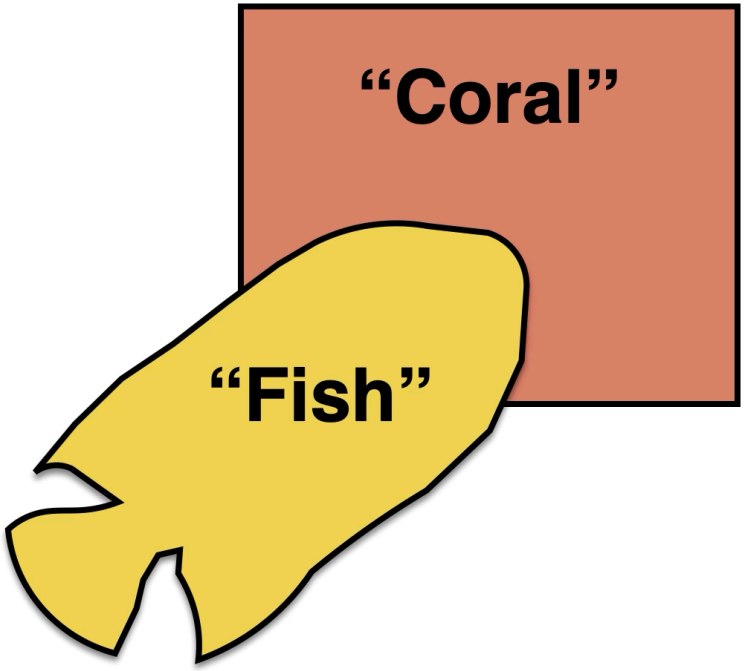
Outline

- Neural networks are *representation* learners
- Auto-encoders and self-supervised learning
- Word embeddings
- Multi-modality

humans also learn representations



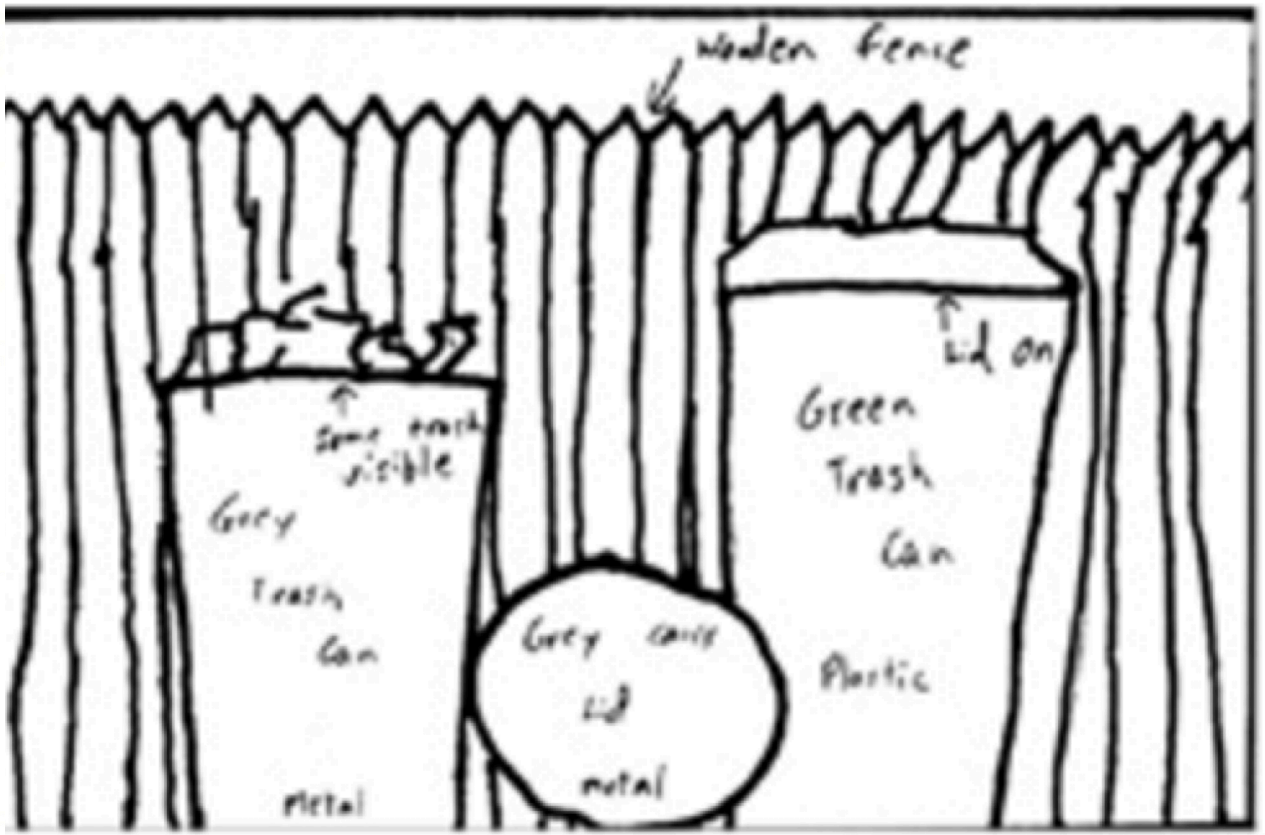




Observed image



Drawn from memory



[Bartlett, 1932]
[Intraub & Richardson, 1989]







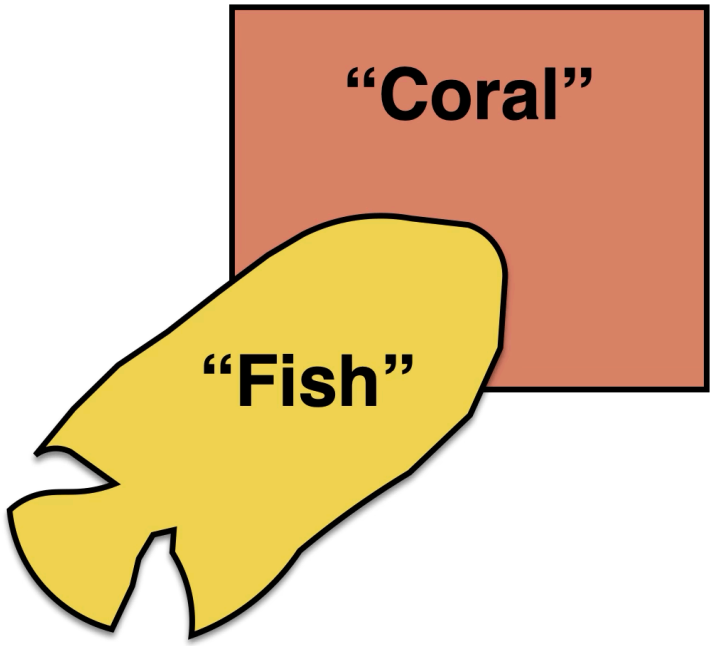






[https://www.behance.net/gallery/35437979/Velocipedia]

Good representations are:



Auto-encoders explicitly aim for

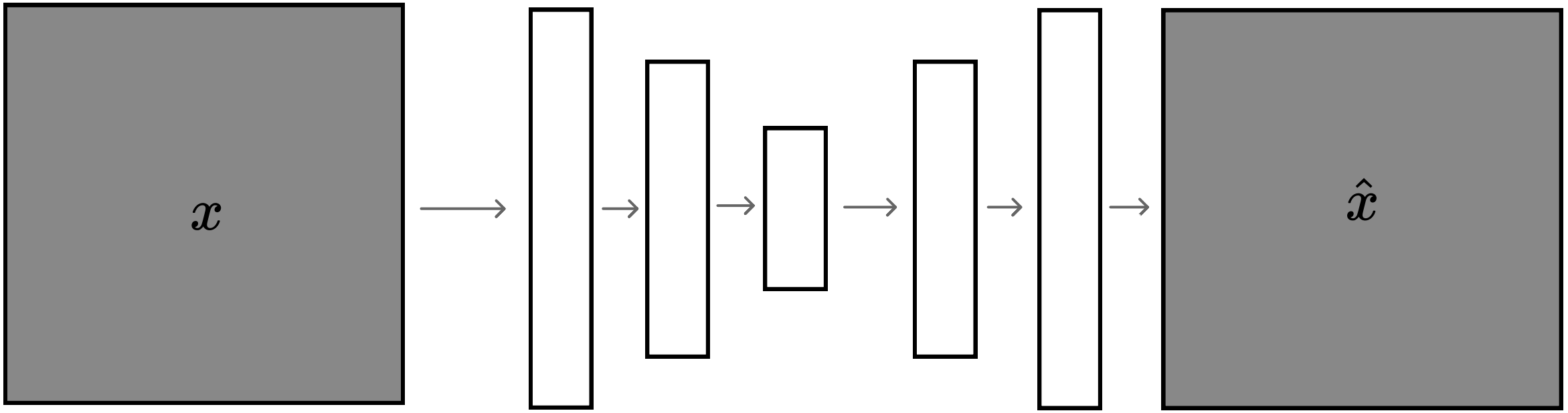
- Compact (*minimal*)
- Explanatory (*roughly sufficient*)

these may just emerge as well

- Disentangled (*independent* factors)
- Interpretable (understandable by humans)
- Make *subsequent problem solving easy*

[See "Representation Learning", Bengio 2013, for more commentary]

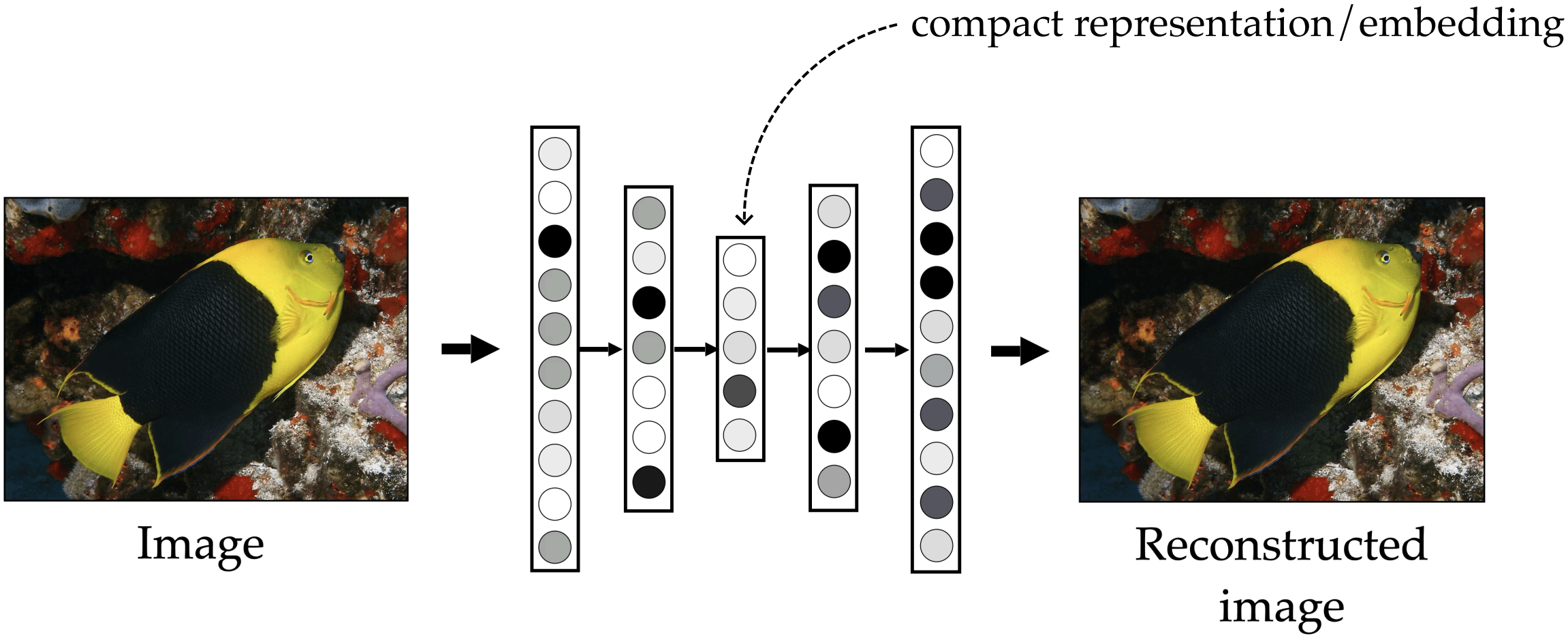
Feature reconstruction



Features

Reconstructed Features

Auto-encoder



"*What I cannot create, I do not understand*." Feynman

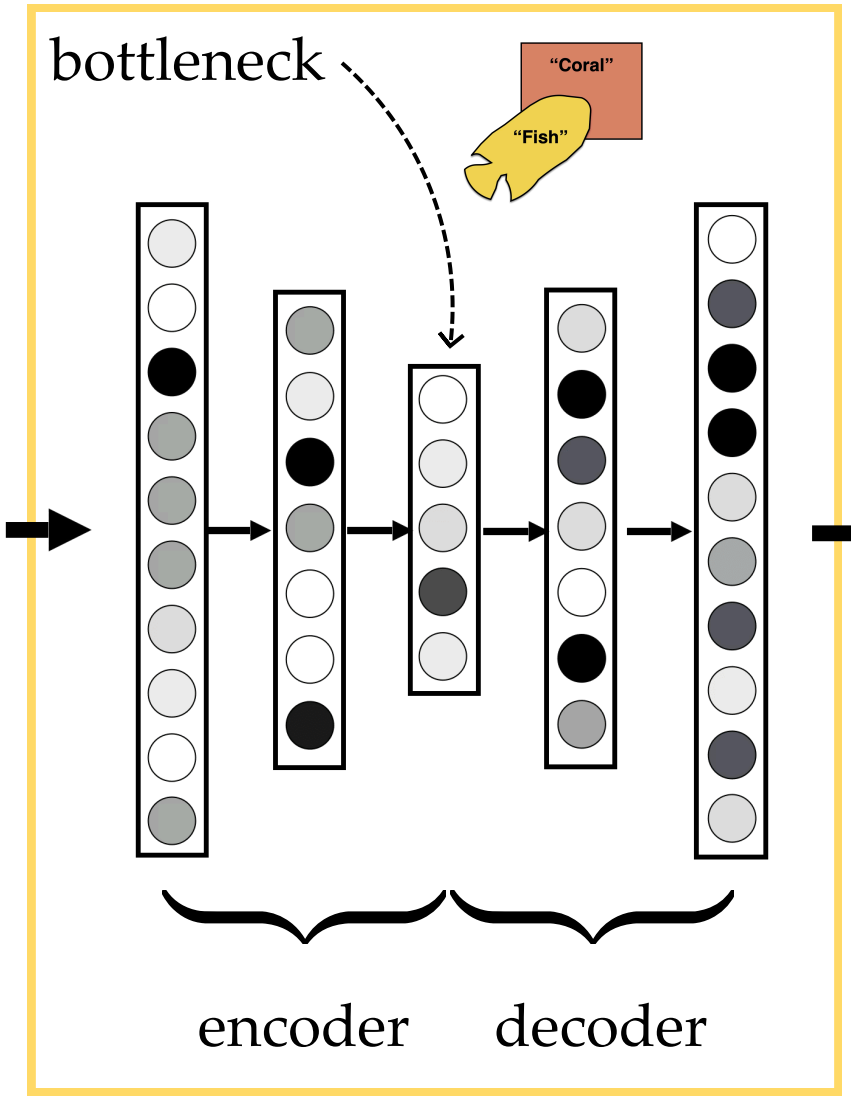
Auto-encoder

$$\min_W ||x - \tilde{x}||^2$$

x



Image



$\tilde{x} = \text{NN}(x; W)$



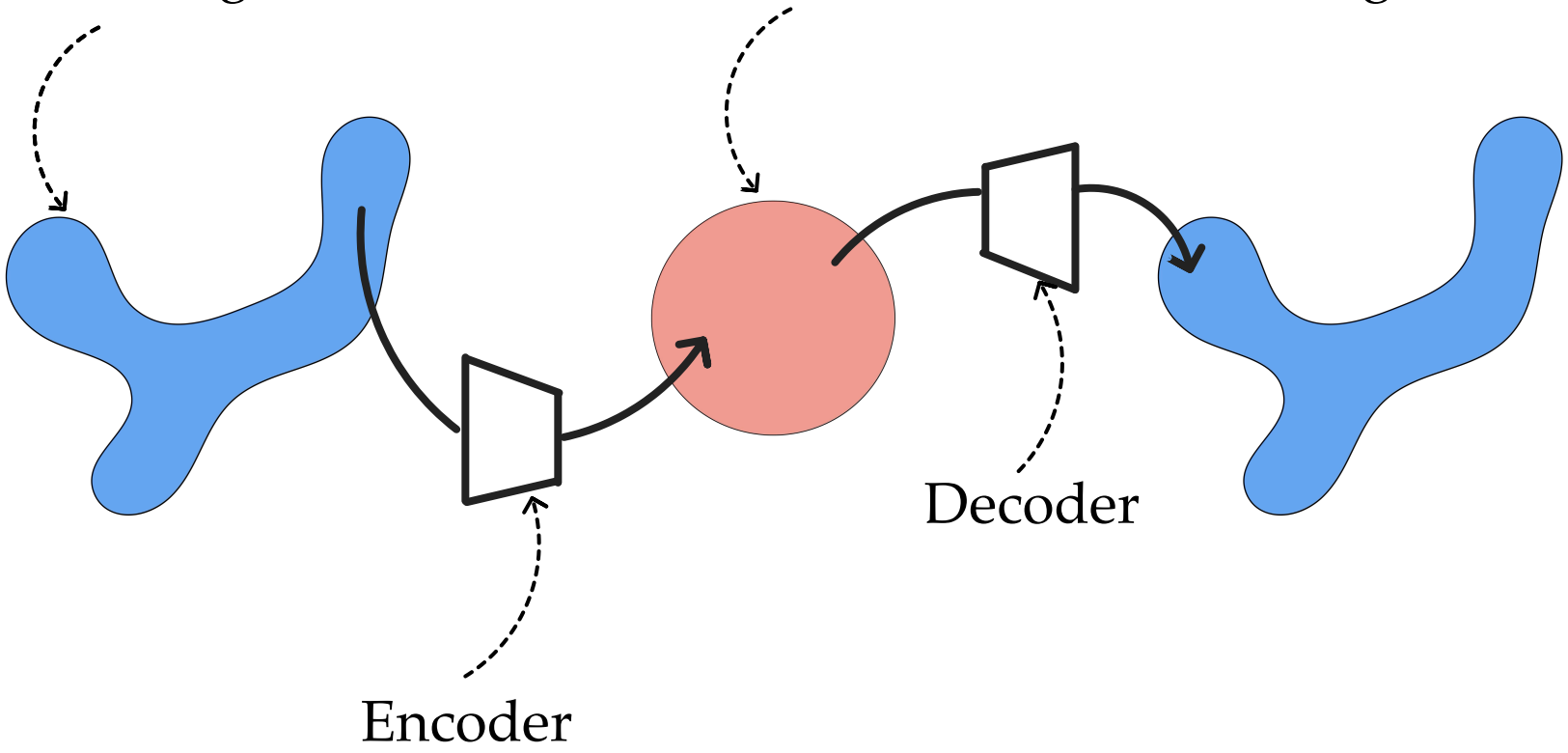
Reconstructed image

Data space:

(high dimensional, irregular)

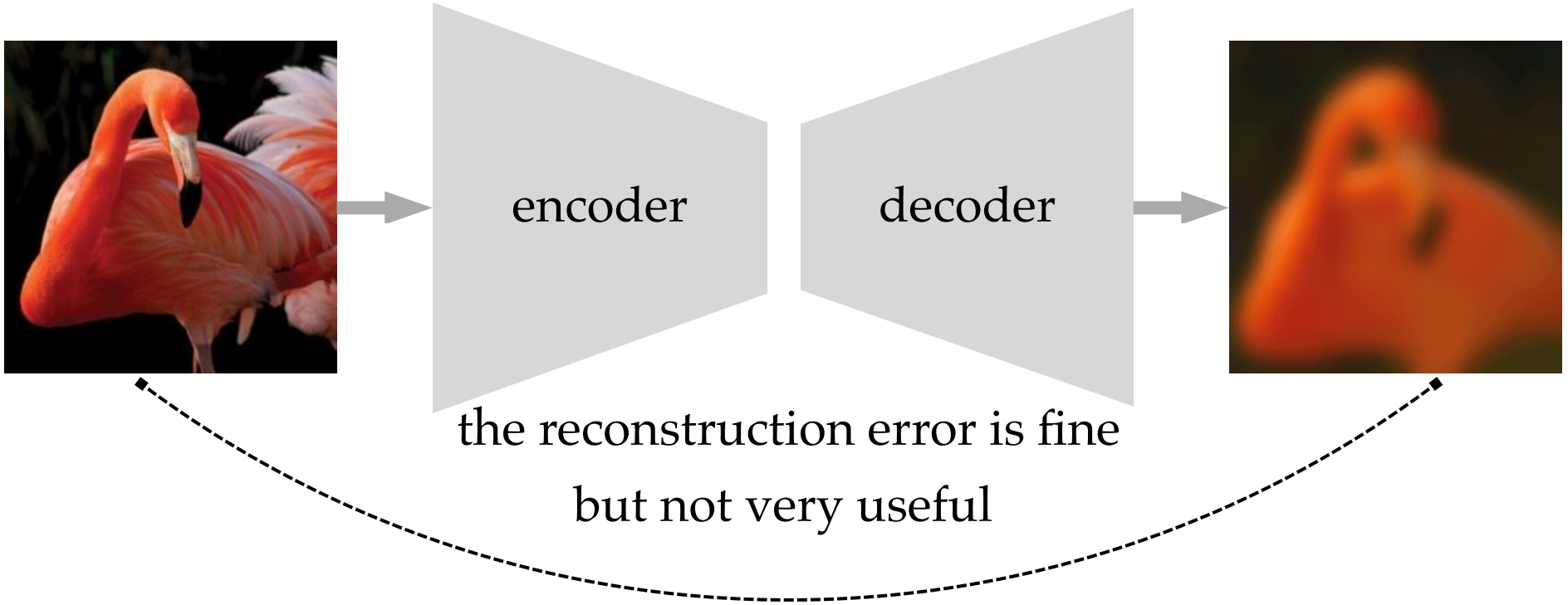
Representation space

(low dimensional, regular)



[Typically, encoders can serve as a translator to get "good representations", whereas decoders can serve as "generative models"]

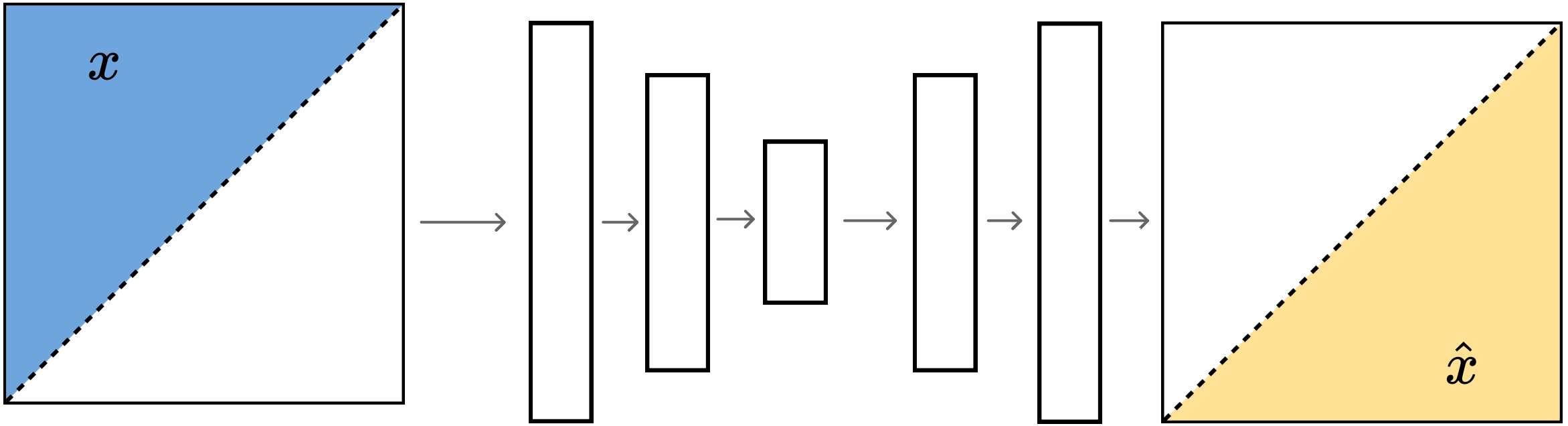
It can be easy to "cheat" with auto-encoders



[Vincent et al, Extracting and composing robust features with denoising autoencoders, ICML 08]

[Steck 20, Zhang et al 17]

Partial feature reconstruction

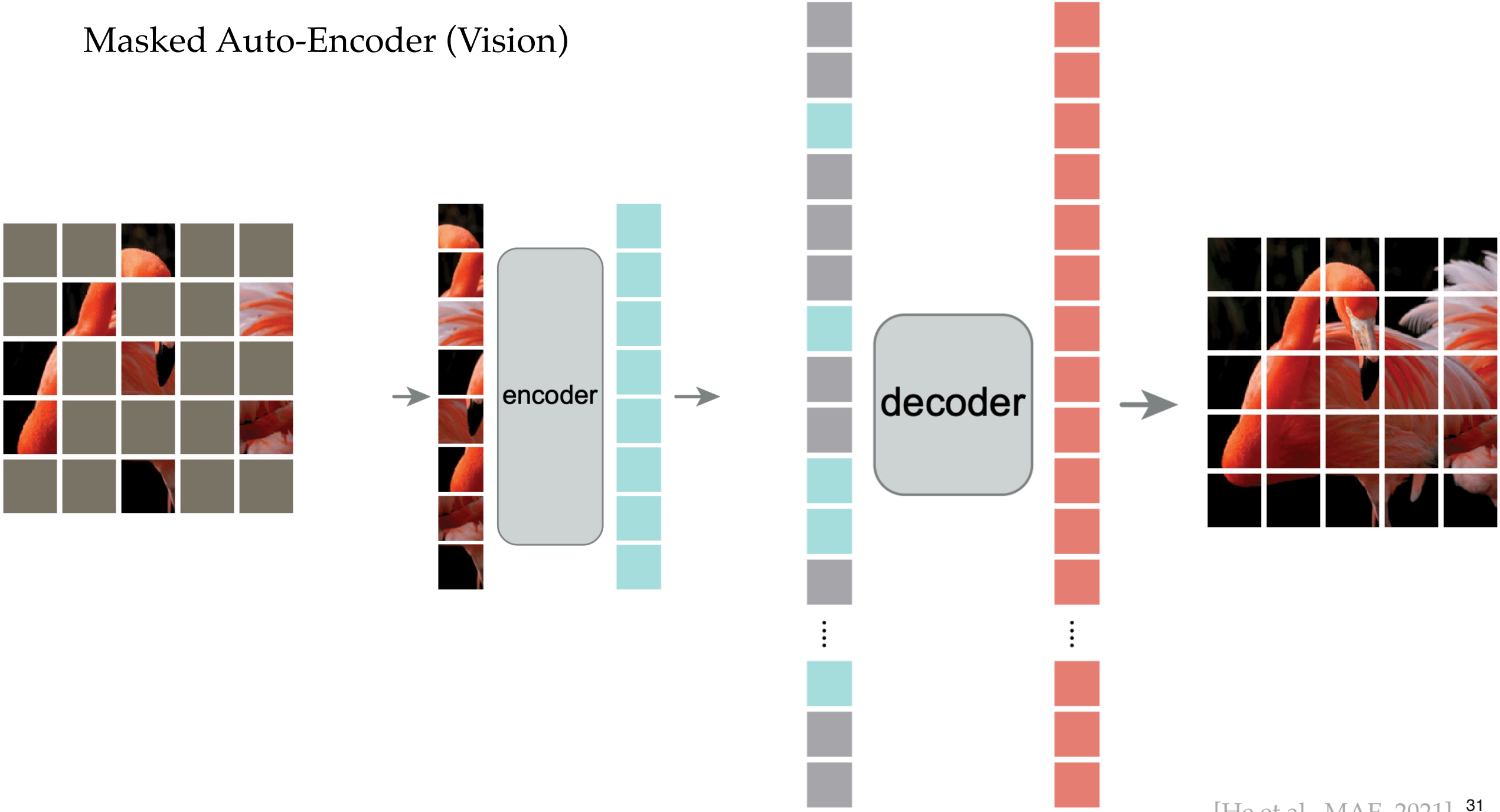


Partial Features

Other Partial Features

Hard reconstruction forces understanding.

Masked Auto-Encoder (Vision)



[He et al., MAE, 2021]

Masking channels

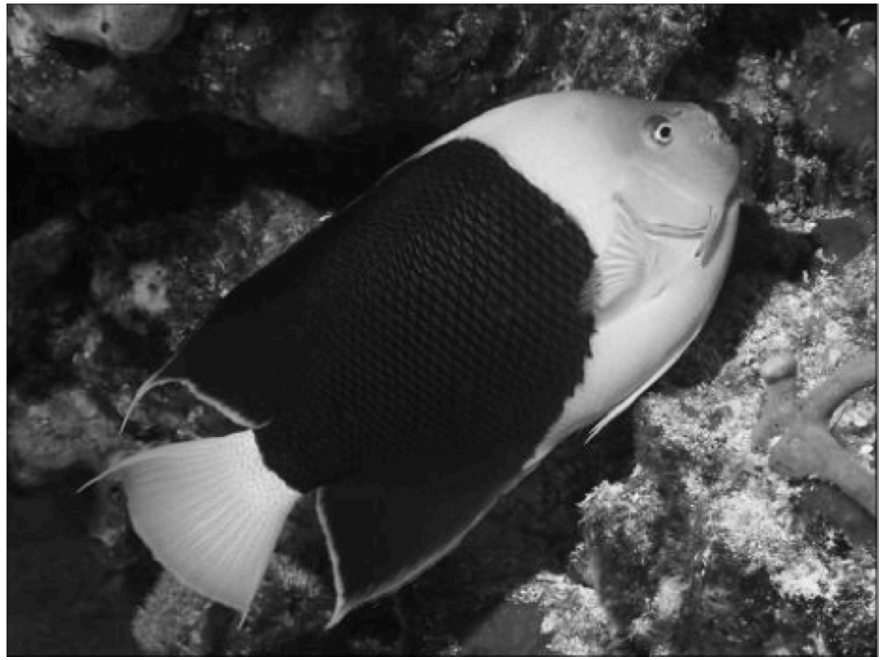
e.g. masking the color

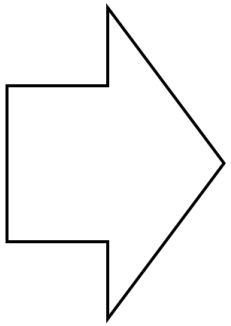


[Zhang, Isola, Efros, ECCV 2016]

Masking channels

predict color from gray-scale

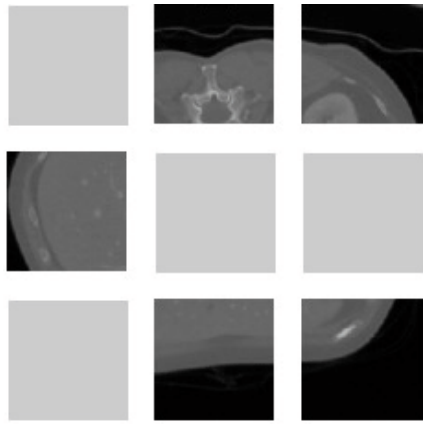
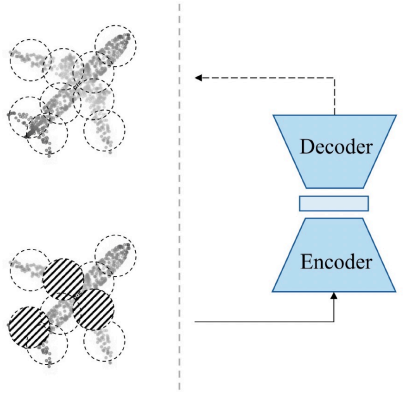
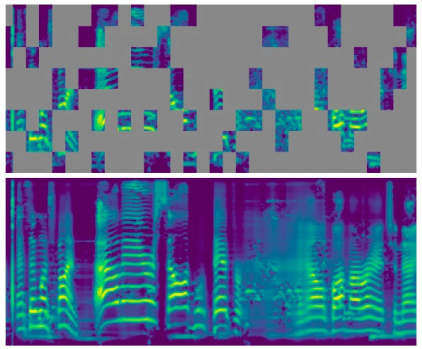
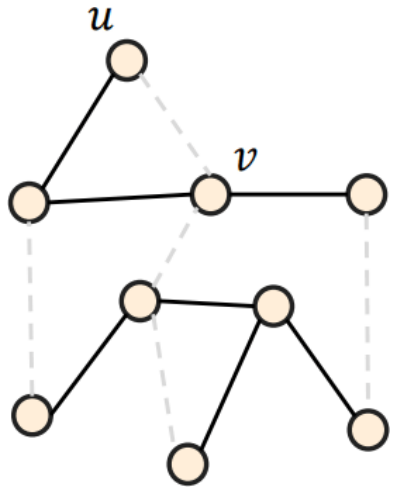
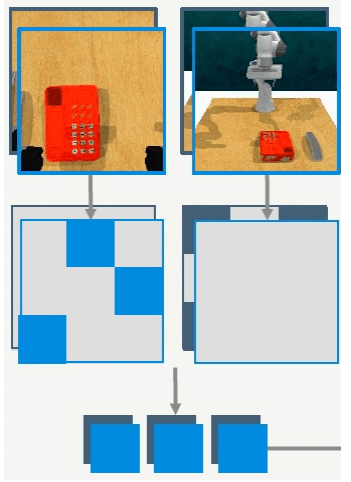




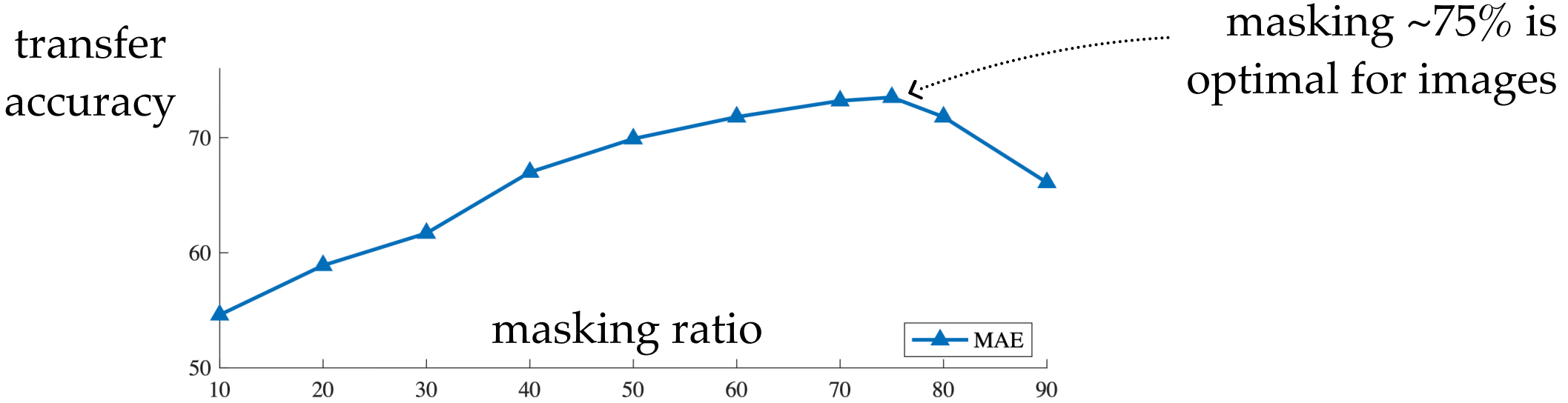


[Zhang, Isola, Efros, ECCV 2016]

Reconstruction works across domains

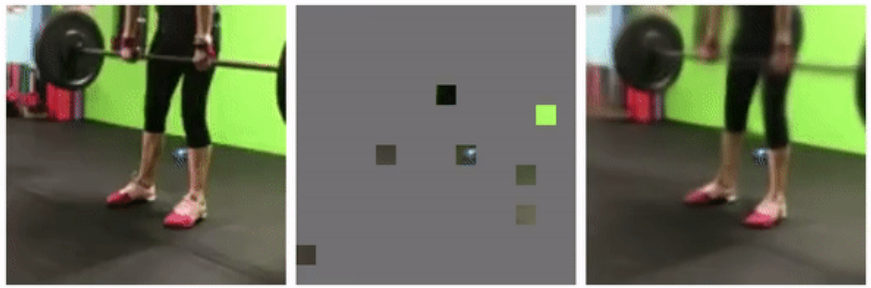
Medical imaging	3D geometry	Audio	Graphs	Robotics
				
scan patches	points or surfaces	spectrogram patches	links or node features	camera frames
[Zhou et al., 2022]	[Point-MAE, ECCV 2022]	[Audio-MAE, NeurIPS 2022]	[MGAE, 2022]	[MV-MWM, ICML 2023]

How much should we hide?



Similar empirical studies show ~15% as optimal for languages, and ~95% for videos

95% masked



[BERT, NAACL 2019 · MAE, CVPR 2022 · MAE-ST, NeurIPS 2022 · VideoMAE, NeurIPS 2022]

How Much Information is the Machine Given during Learning?

- **"Pure" Reinforcement Learning (cherry)**
  - The machine predicts a scalar reward given once in a while.
  - **A few bits for some samples**
- **Supervised Learning (icing)**
  - The machine predicts a category or a few numbers for each input
  - Predicting human-supplied data
  - **10→10,000 bits per sample**
- **Self-Supervised Learning (cake génoise)**
  - The machine predicts any part of its input for any observed part.
  - Predicts future frames in videos
  - **Millions of bits per sample**



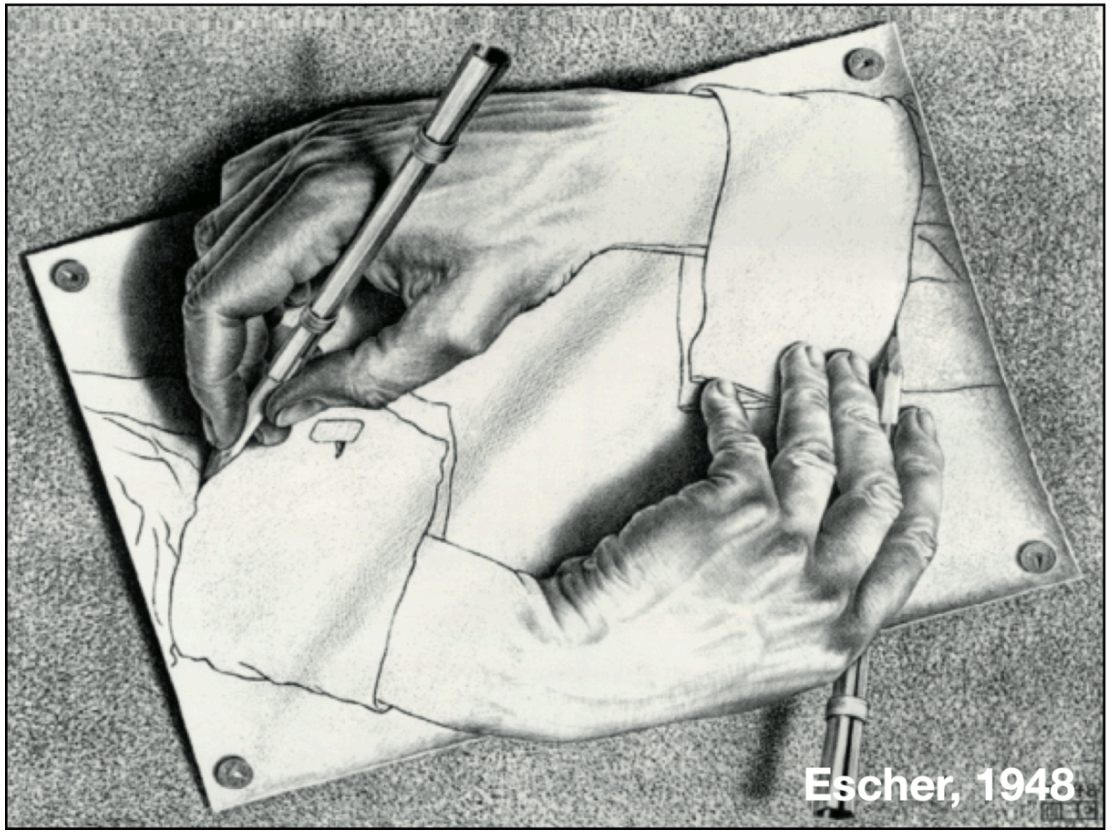
© 2019 IEEE International Solid-State Circuits Conference

[Slide Credit: Yann LeCun]

Outline

- Neural networks are *representation* learners
- Auto-encoders and self-supervised learning
- Word embeddings
- Multi-modality

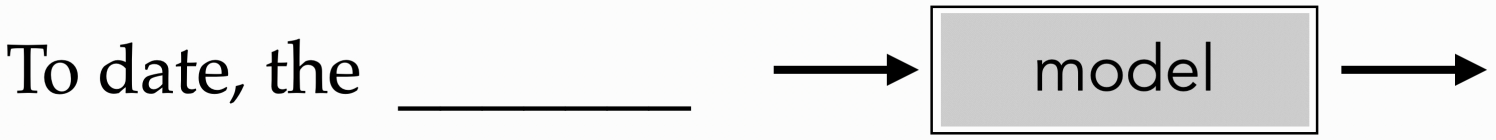
Large Language Models (LLMs) are trained in this self-supervised way:



- Scrape the internet for plain texts.
- Cook up “labels” from these texts.
- Use the unhidden part to predict the hidden label.

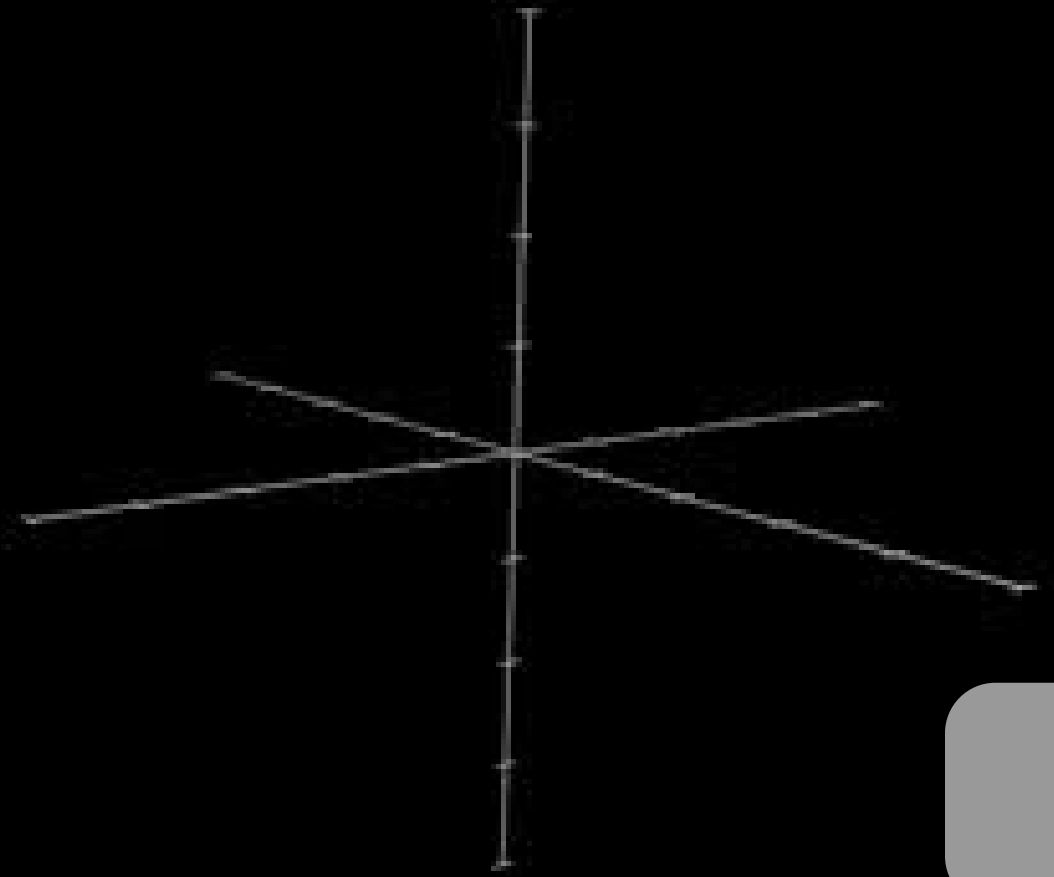
"To date, the cleverest thinker of all time was Isaac."

feature	label
⋮	⋮
To date, the	cleverest
To date, the cleverest	thinker
To date, the cleverest thinker	was
⋮	⋮
To date, the cleverest thinker of all time was	Isaac



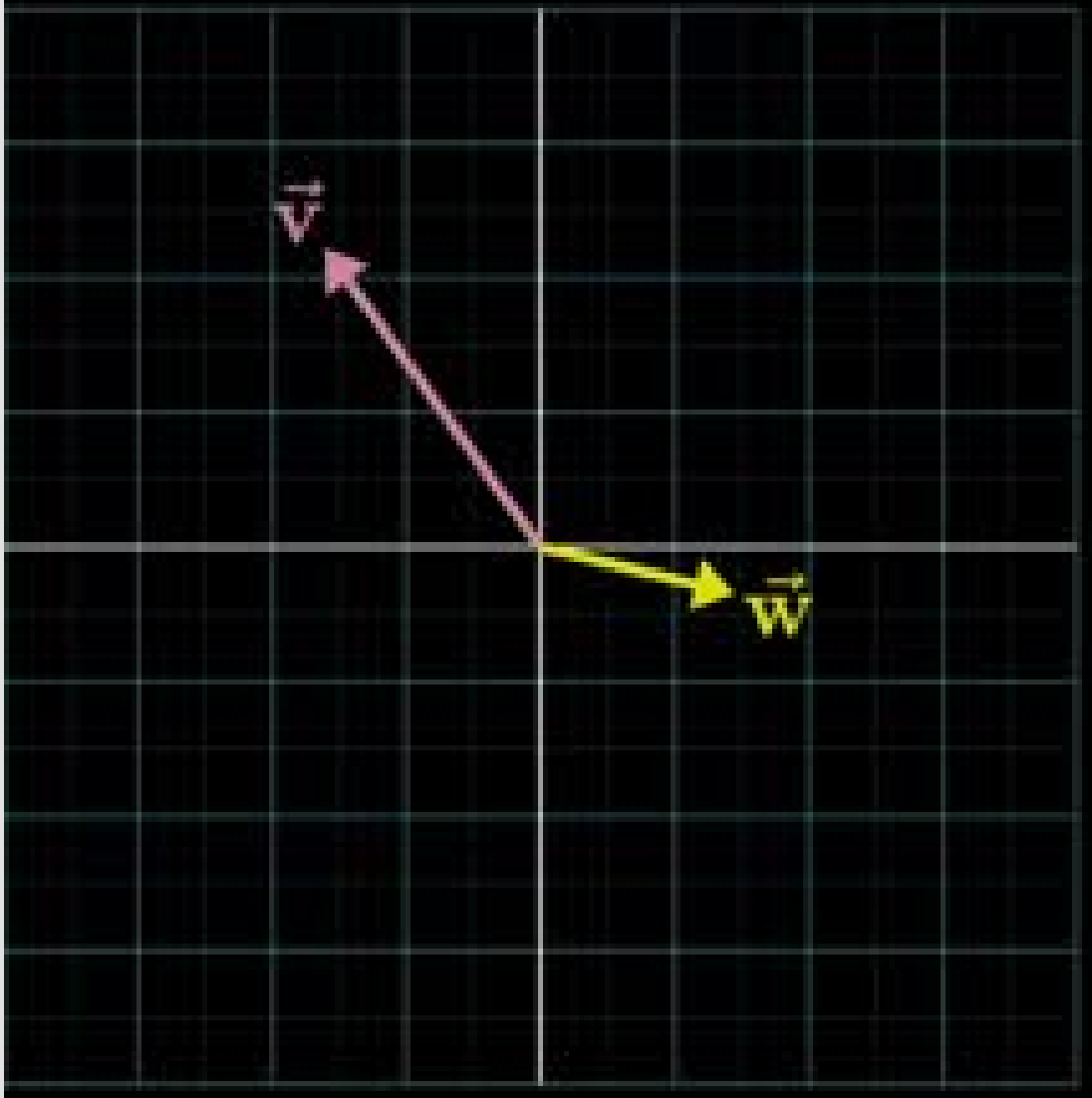
Word embedding

Words → Vectors



[video edited from 3b1b]

dot-product similarity



$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_n \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_n \end{bmatrix} = \begin{matrix} v_1 w_1 \\ + \\ v_2 w_2 \\ + \\ v_3 w_3 \\ + \\ \vdots \\ + \\ v_n w_n \\ \| \\ -3.11 \end{matrix}$$

Dot product

A good word embedding space is equipped with sensible dot-product similarity

For now, let's look at how good embeddings enable "soft" dictionary look-up:

```
dict_en2fr = {
  "apple" : "pomme",
  "banana": "banane",
  "lemon" : "citron"}
```

Key Value

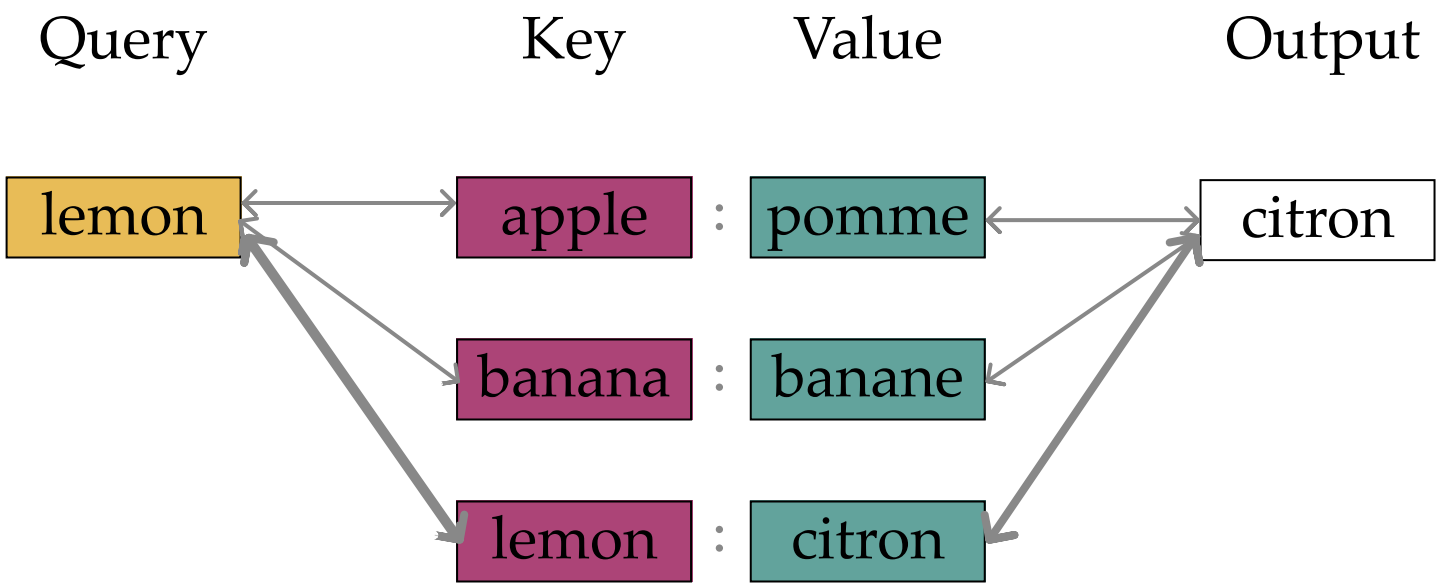
apple : pomme

banana : banane

lemon : citron

```python
dict_en2fr = {
  "apple" : "pomme",
  "banana": "banane",
  "lemon" : "citron"}

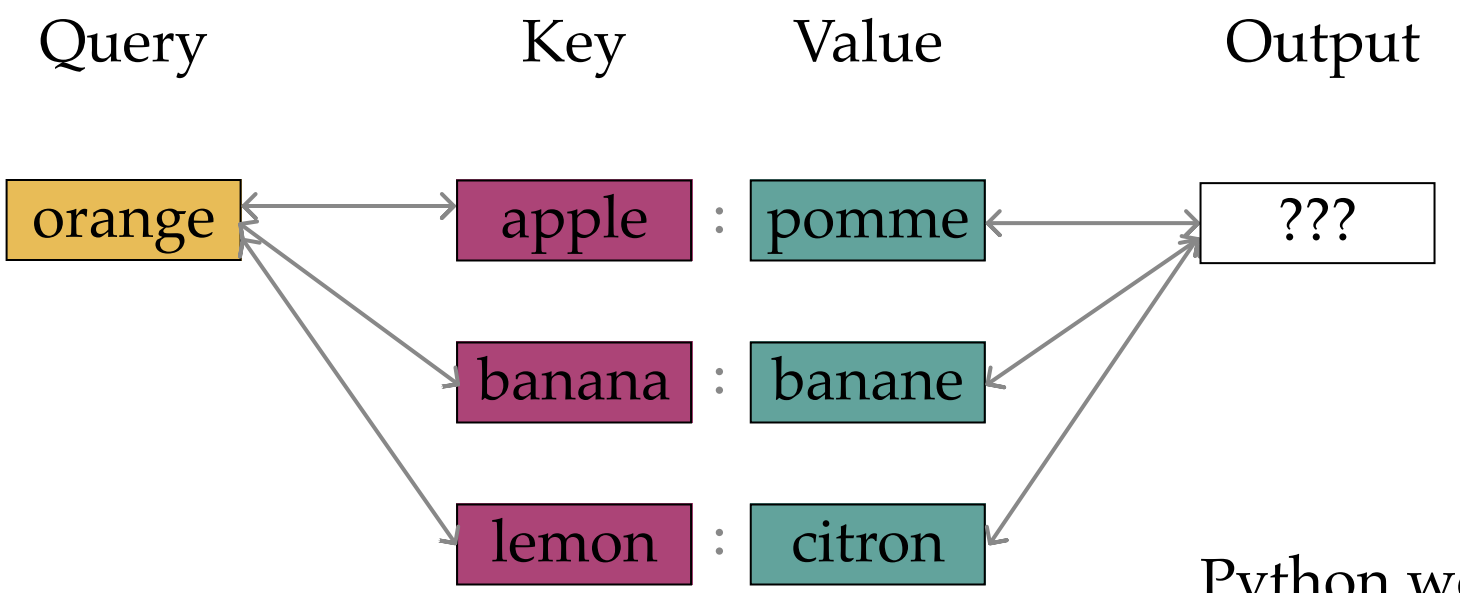
query = "lemon"
output = dict_en2fr[query]
```



What if we run

```
dict_en2fr = {
  "apple" : "pomme",
  "banana": "banane",
  "lemon" : "citron"}

query = "orange"
output = dict_en2fr[query]
```



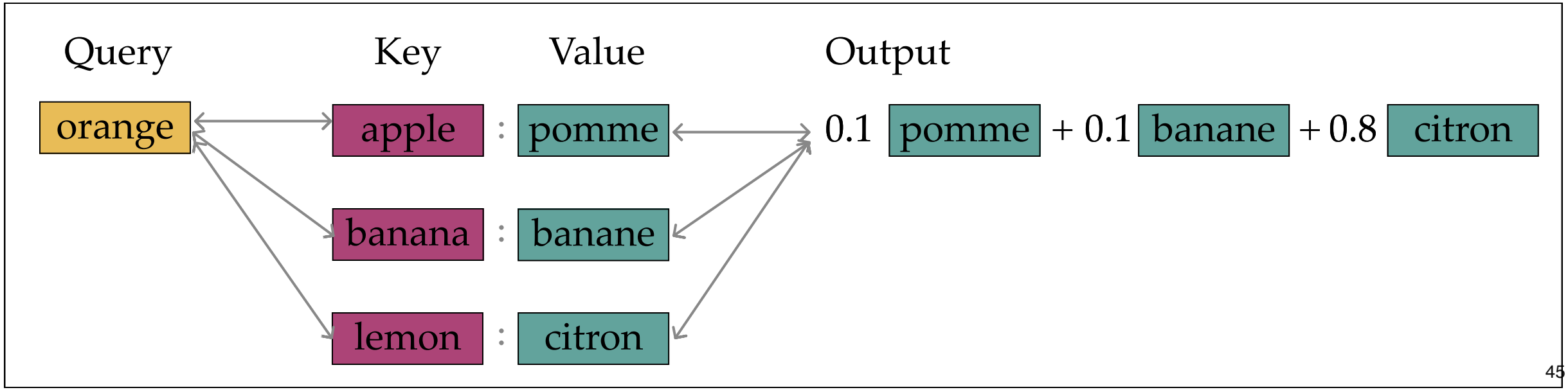
Python would complain. 🤯

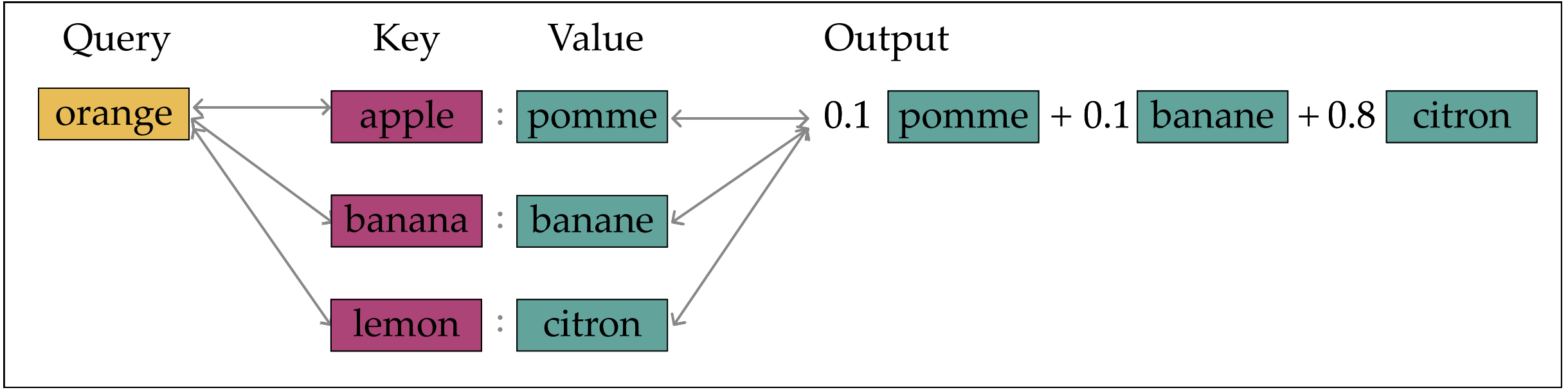
What if we run

```
dict_en2fr = {
  "apple" : "pomme",
  "banana": "banane",
  "lemon" : "citron"}

query = "orange"
output = dict_en2fr[query]
```

But we can probably see the rationale behind this:





We put (query, key, value) in "good" embeddings in our human brains

such that "merging" the values 0.1 pomme + 0.1 banane + 0.8 citron

via these merging percentages [0.1 0.1 0.8] made sense.

Very roughly, the attention mechanism in transformers automates this process. We'll see the details next week.

Outline

- Neural networks are *representation* learners
- Auto-encoders and self-supervised learning
- Word embeddings
- Multi-modality

Plato's cave



[images credit: visionbook.mit.edu · top right: Plato's cave, Saenredam, 1604]

Contrastive learning: SimCLR

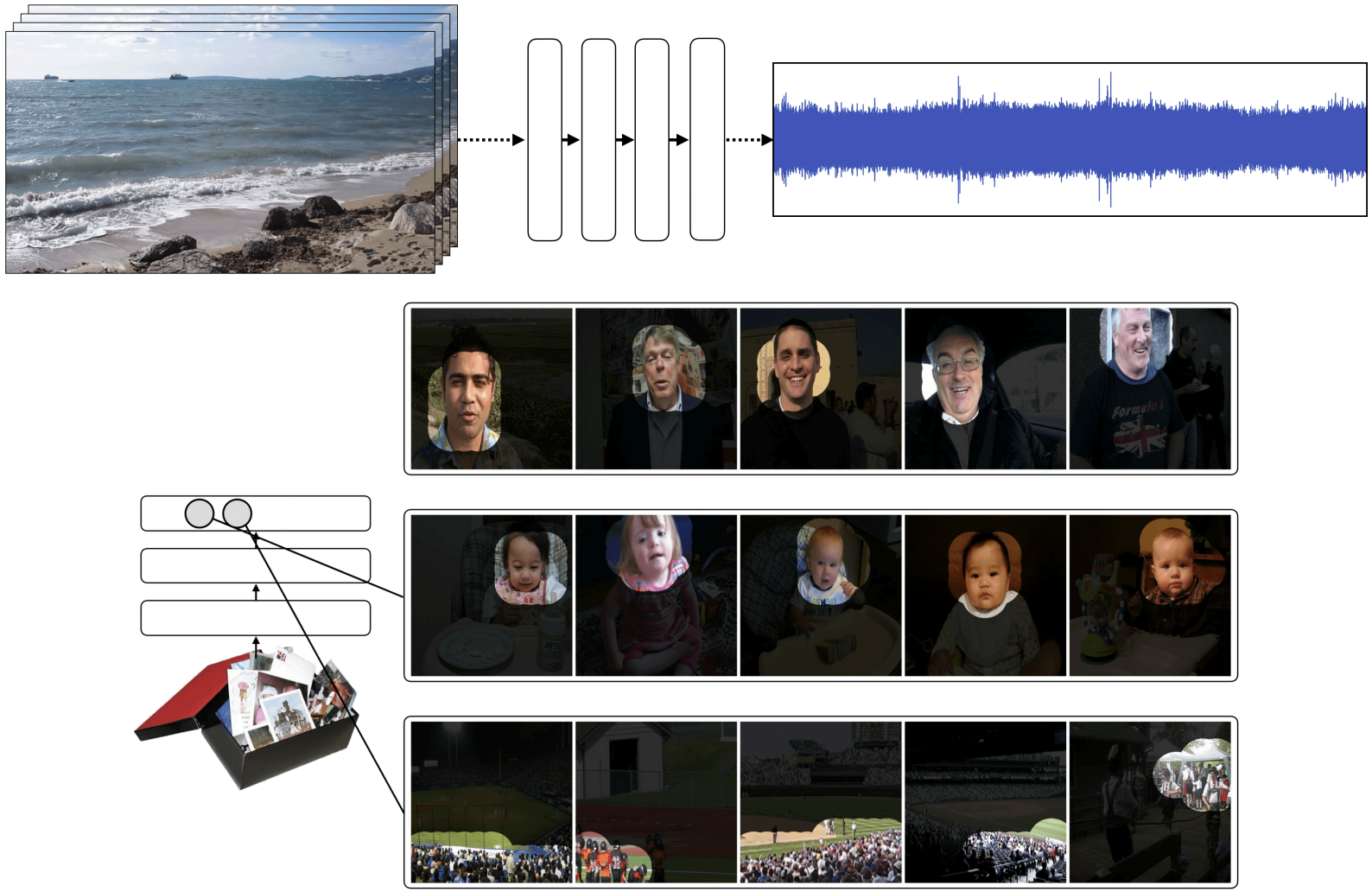
We take two random crops of the same image, with colors changed, and pull their representations together.

Crops from different images are pushed apart.

What we change, such as color, is what the representation learns to ignore.

[Chen, Kornblith, Norouzi & Hinton · SimCLR · ICML 2020]

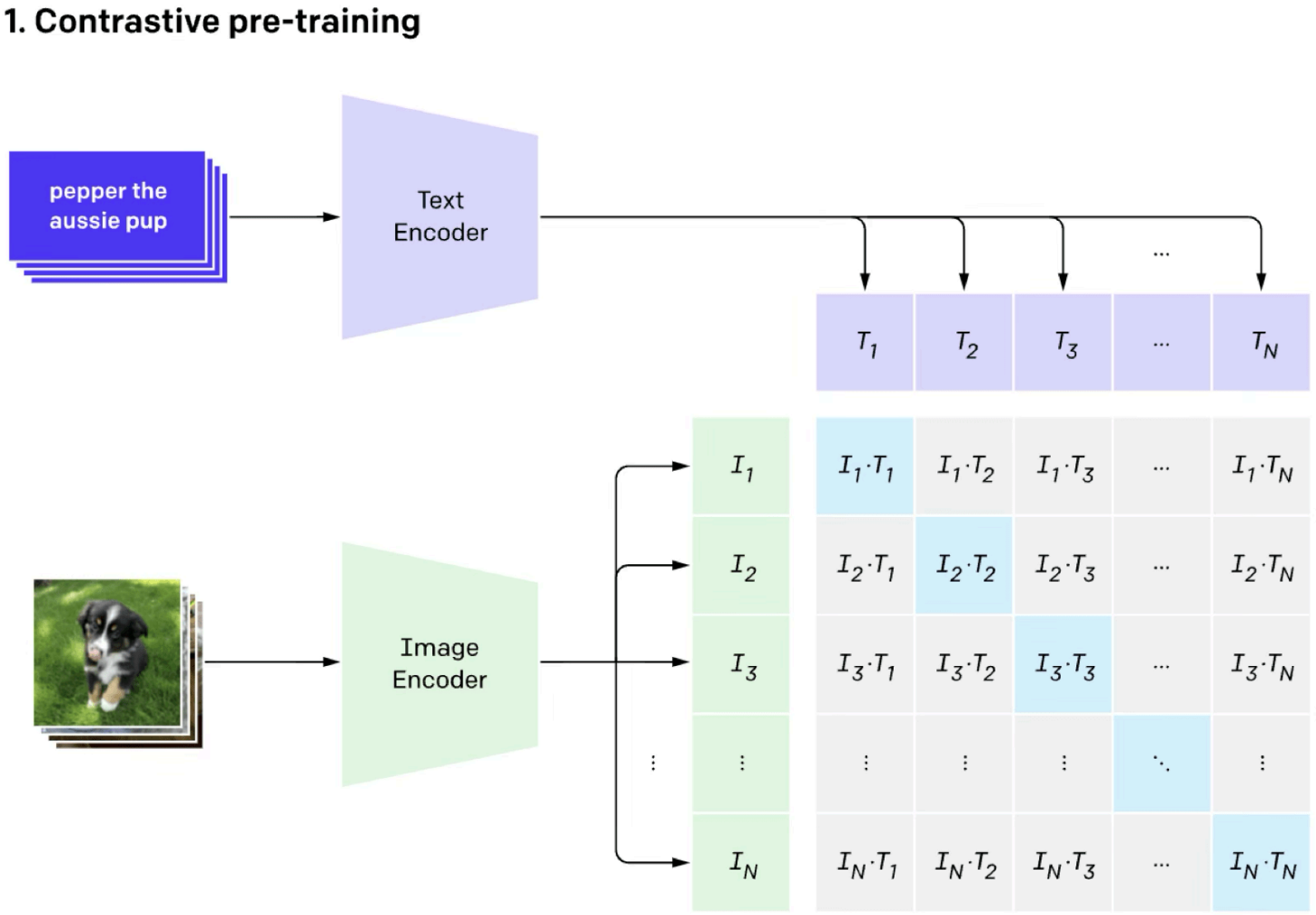
Predicting sound from video



Units that learned to fire on faces, babies, and crowds: things that make sound.

[Owens et al., Ambient Sound Provides Supervision for Visual Learning, ECCV 2016]

CLIP: images and captions in one space



Two encoders map images and captions into one space.

Training raises the score of each image with its own caption.

To classify a new image, we compare it with prompts like “a photo of a dog”. We will try this in the lab.

[Radford et al. · CLIP · ICML 2021]

e.g. DALL·E 2: text-to-image generation

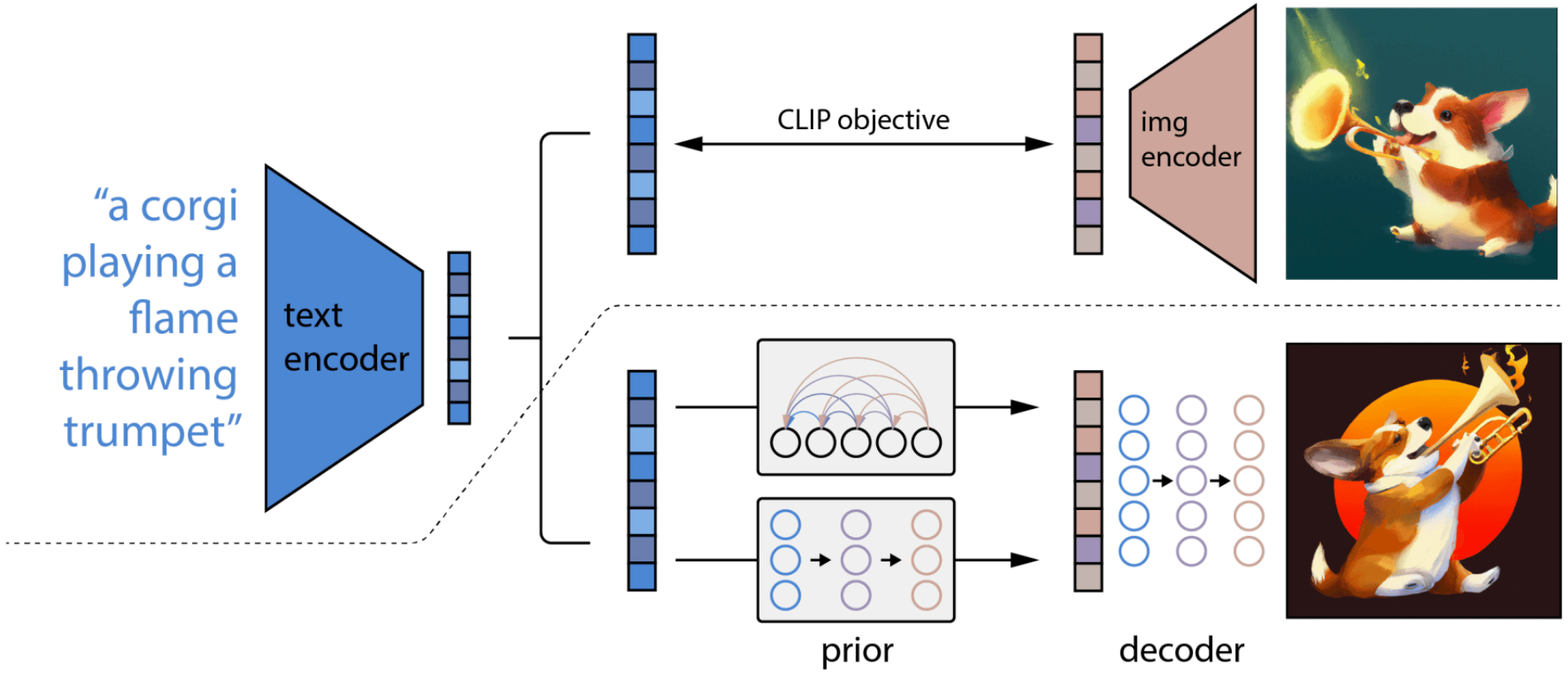
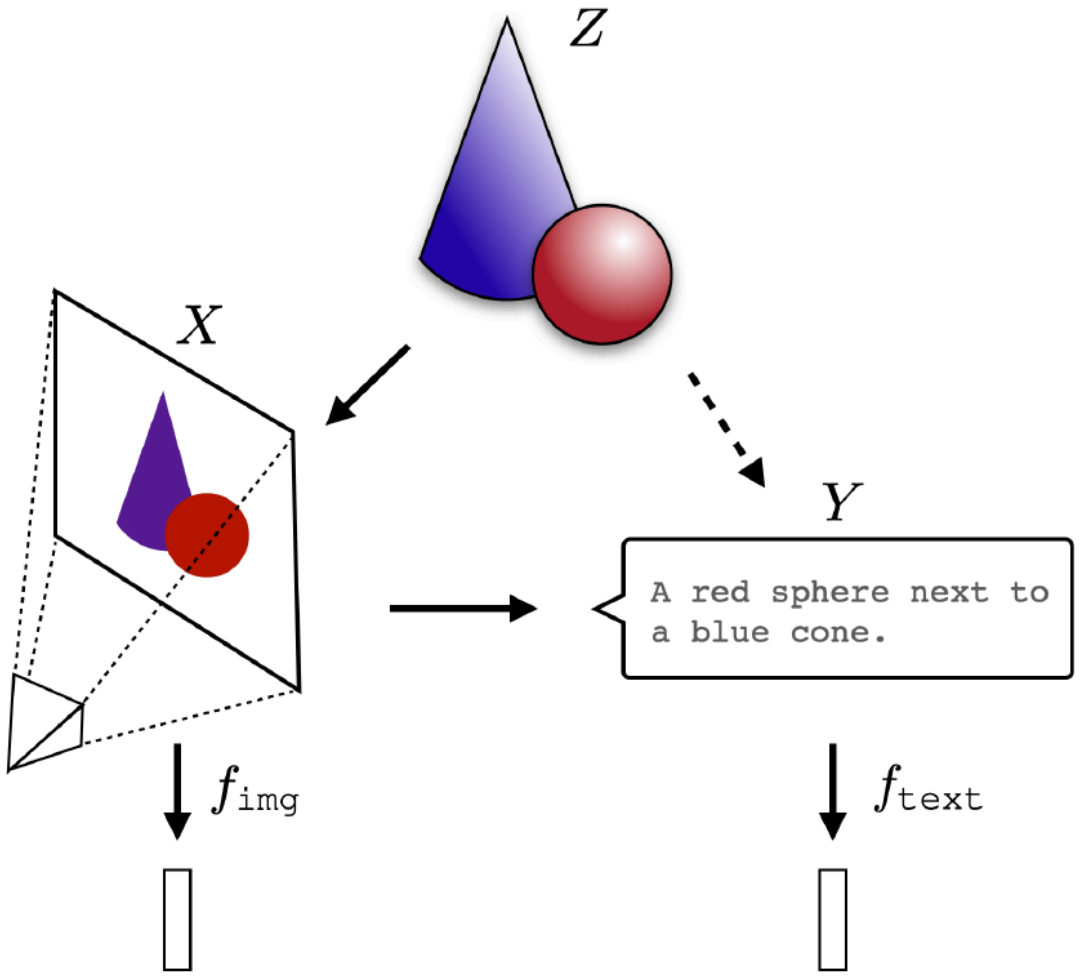


Figure 2: A high-level overview of unCLIP. Above the dotted line, we depict the CLIP training process, through which we learn a joint representation space for text and images. Below the dotted line, we depict our text-to-image generation process: a CLIP text embedding is first fed to an autoregressive or diffusion prior to produce an image embedding, and then this embedding is used to condition a diffusion decoder which produces a final image. Note that the CLIP model is frozen during training of the prior and decoder.

[Ramesh et al., Hierarchical Text-Conditional Image Generation with CLIP Latents, 2022]

Different modalities, converging representations, without explicit alignment



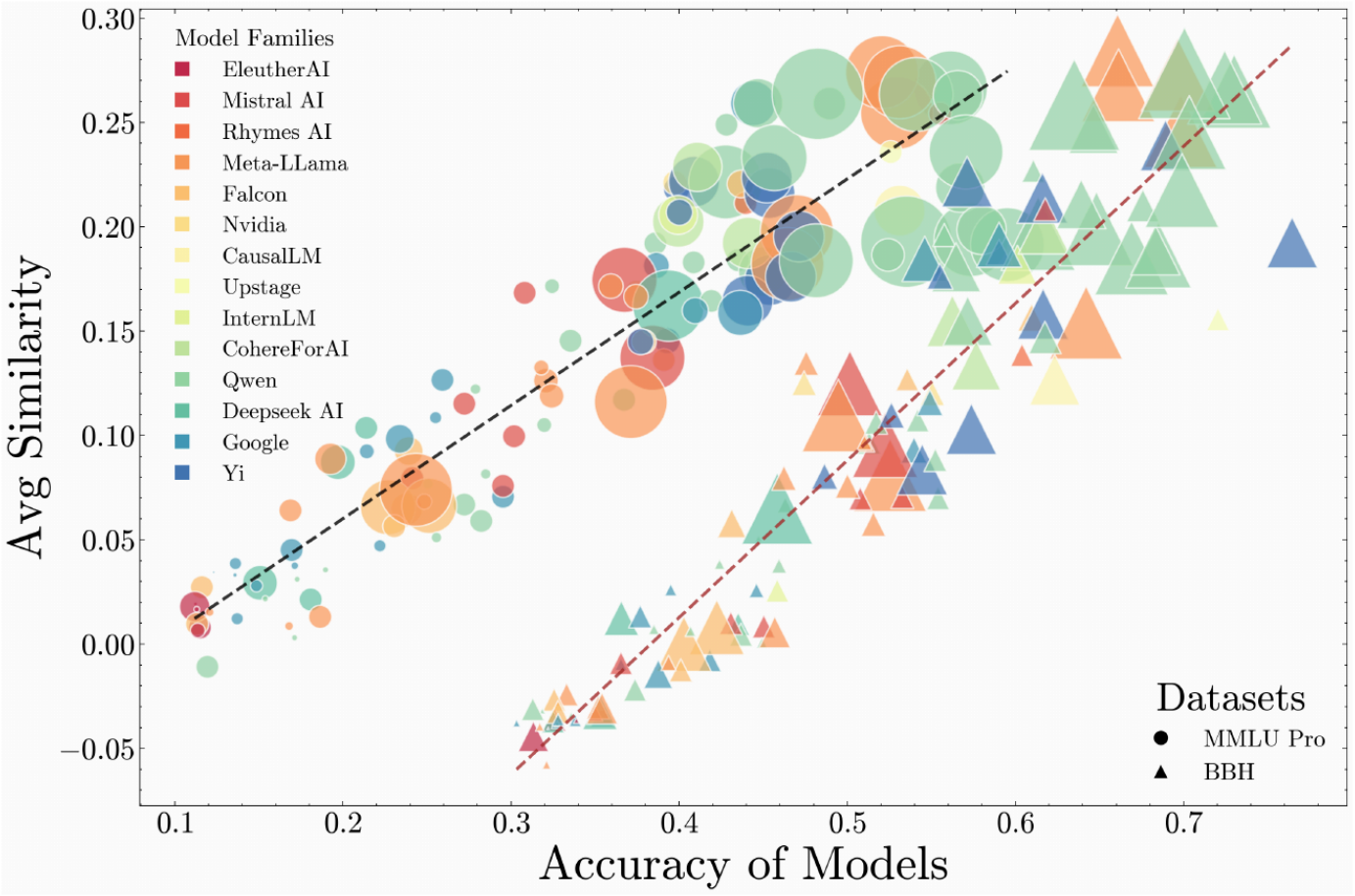
Images X and text Y are two projections of the same world Z.

Vision and language models agree on which things are similar, and stronger models agree more.

Models may be converging toward a shared representation of reality.

[Huh, Cheung, Wang & Isola · The Platonic Representation Hypothesis · ICML 2024 · Fig. 1]

Algorithmic monoculture



The more accurate a language model, the more its mistakes match those of other models, even across companies.

When many decisions rest on similar models, the same people can be failed everywhere.

[Kleinberg & Raghavan, PNAS 2021 · Bommasani et al., NeurIPS 2022 · figure: Goel et al., ICML 2025, Fig. 6]

Summary

- We looked at the mechanics of NNs. Today we see they learn representations, just like our brains do.
- This is useful because representations transfer — they act as prior knowledge that enables quick learning on new tasks.
- Representations can also be learned without labels, e.g. as we do in unsupervised or self-supervised learning. This is great since labels are expensive and limiting.
- Without labels, there are many ways to learn representations:
  - representations as compressed codes, an auto-encoder with a bottleneck
  - representations that are predictive of their context
  - representations that are shared across sensory modalities