Learning with Less Human Supervision

Danna Gurari
University of Colorado Boulder
Spring 2025

https://dannagurari.colorado.edu/course/neural-networks-and-deep-learning-spring-2025/

Review

- Last lecture on Speeding Up Learning and Inference
  - Motivation
  - Hardware tricks
  - Architectural tricks
  - Training tricks
  - Programming tutorial

- Assignments (Canvas)
  - Final project outline due Thursday

- Questions?

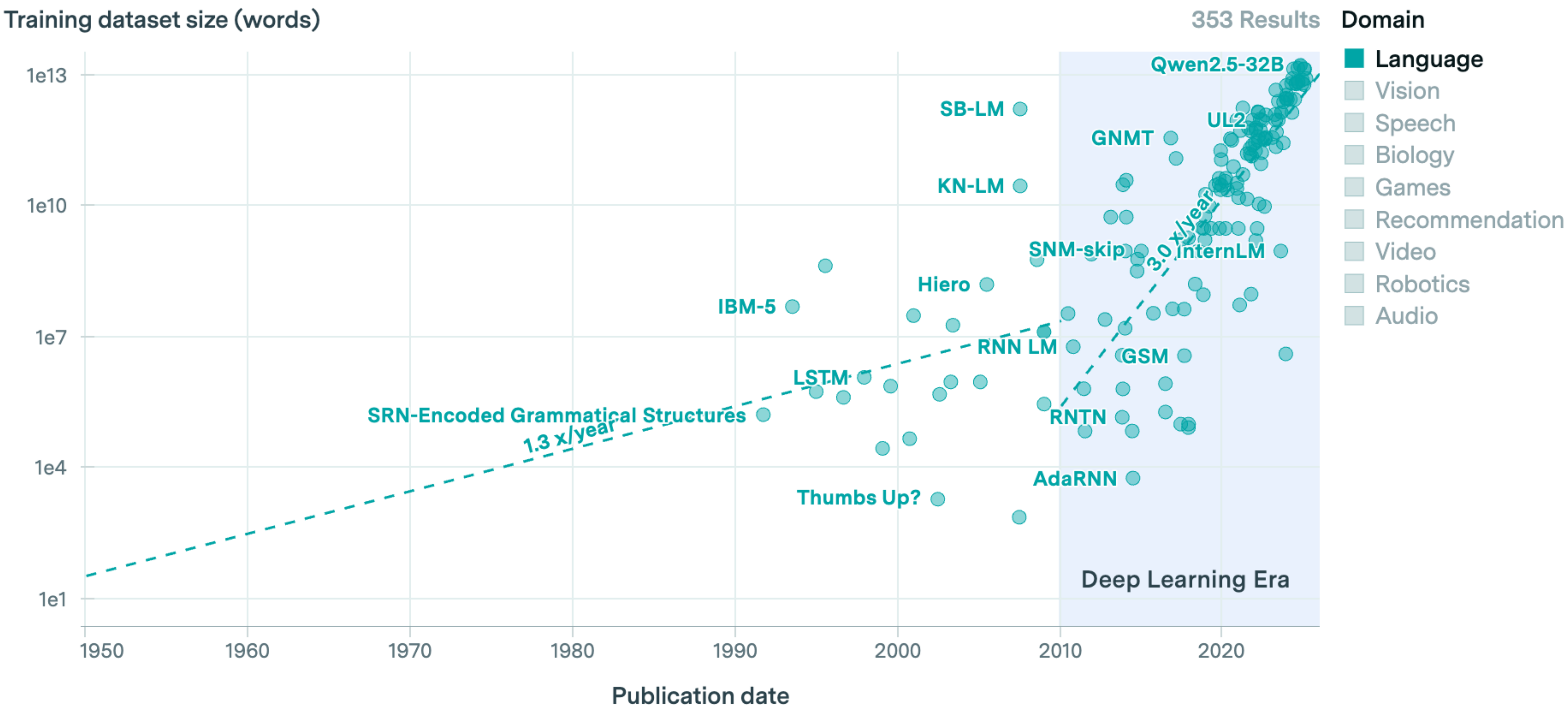
Today’s Topics

- Motivation
- Active learning
- Reinforcement learning
- Self-supervised learning
- And more…

Today’s Topics

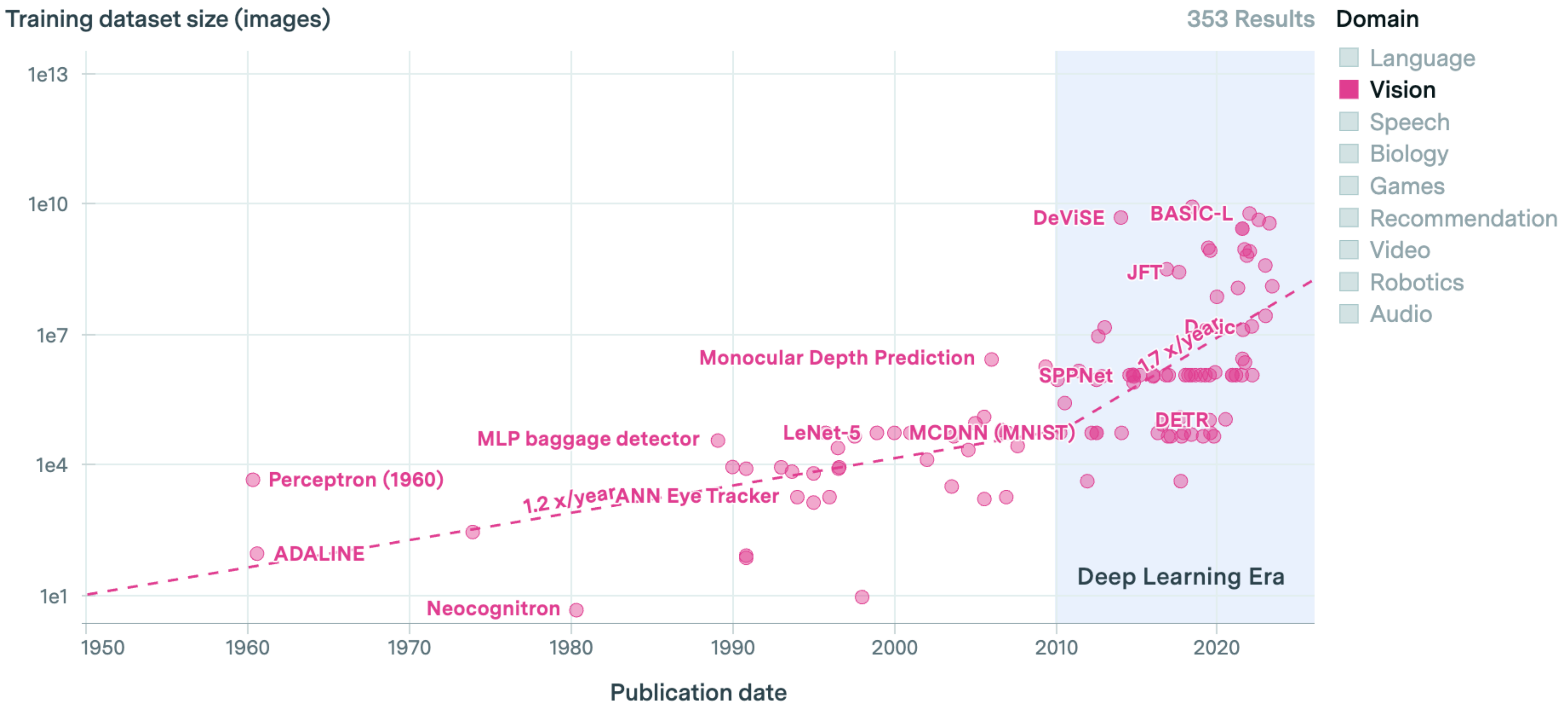
- Motivation
- Active learning
- Reinforcement learning
- Self-supervised learning
- And more…

Trend: More Training Data for Better Models



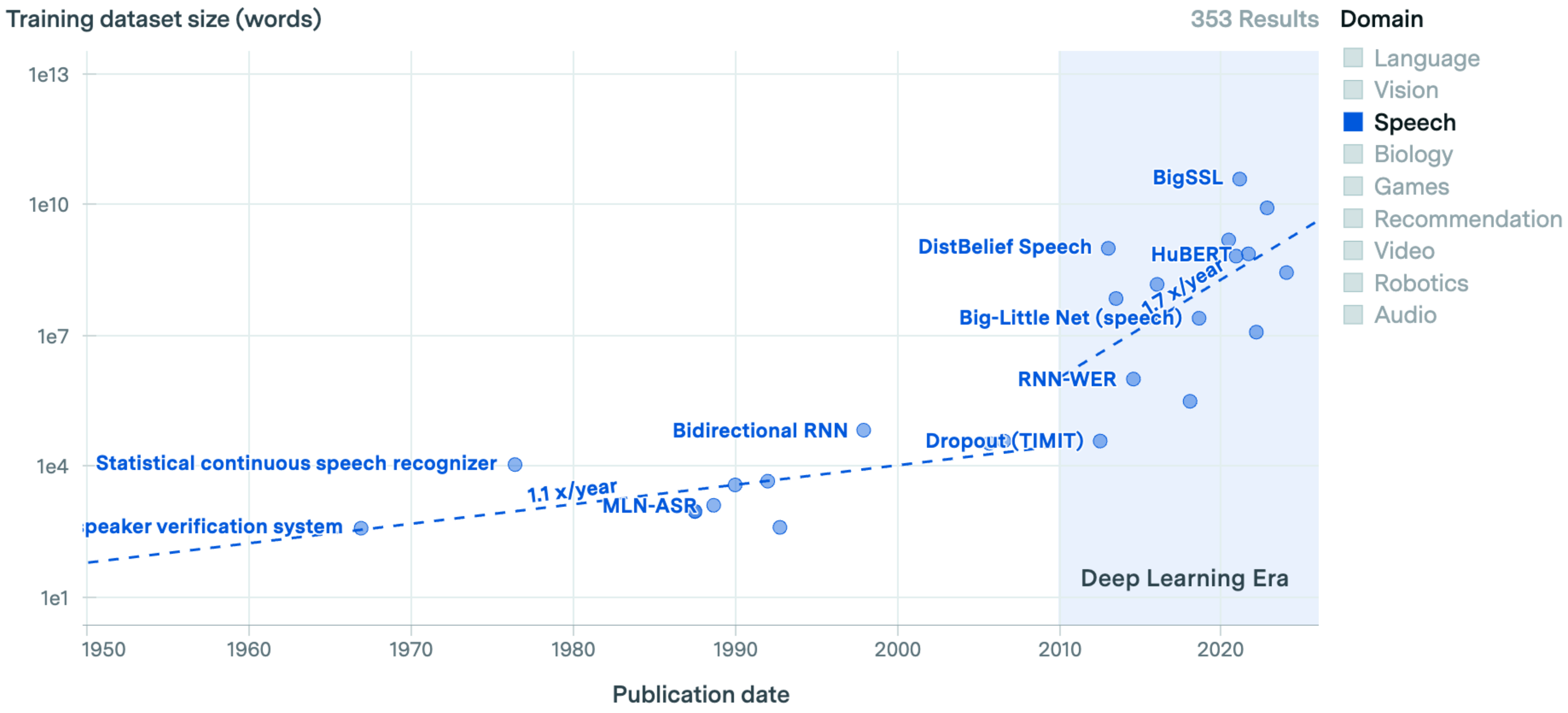
https://epoch.ai/data/notable-ai-models#explore-the-data

Trend: More Training Data for Better Models



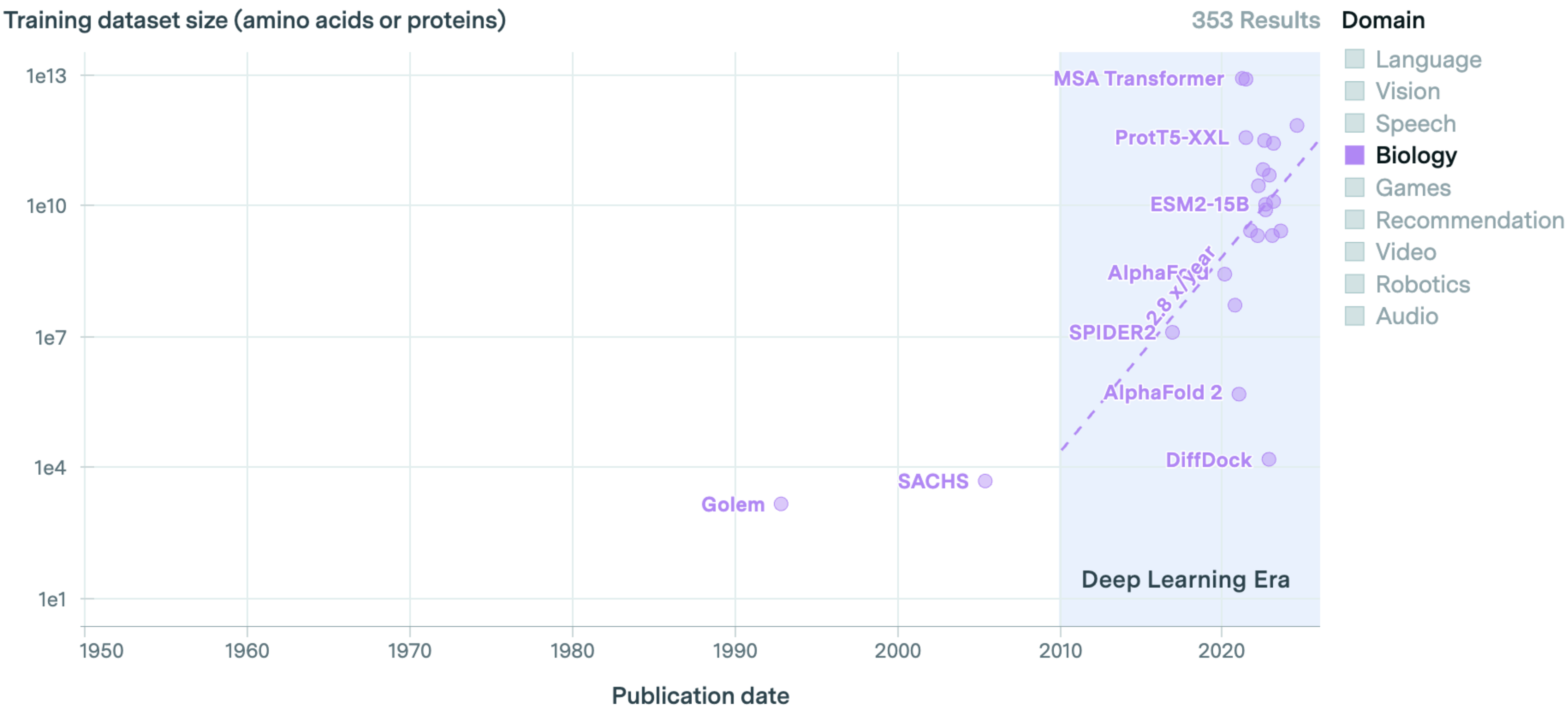
https://epoch.ai/data/notable-ai-models#explore-the-data

Trend: More Training Data for Better Models



https://epoch.ai/data/notable-ai-models#explore-the-data

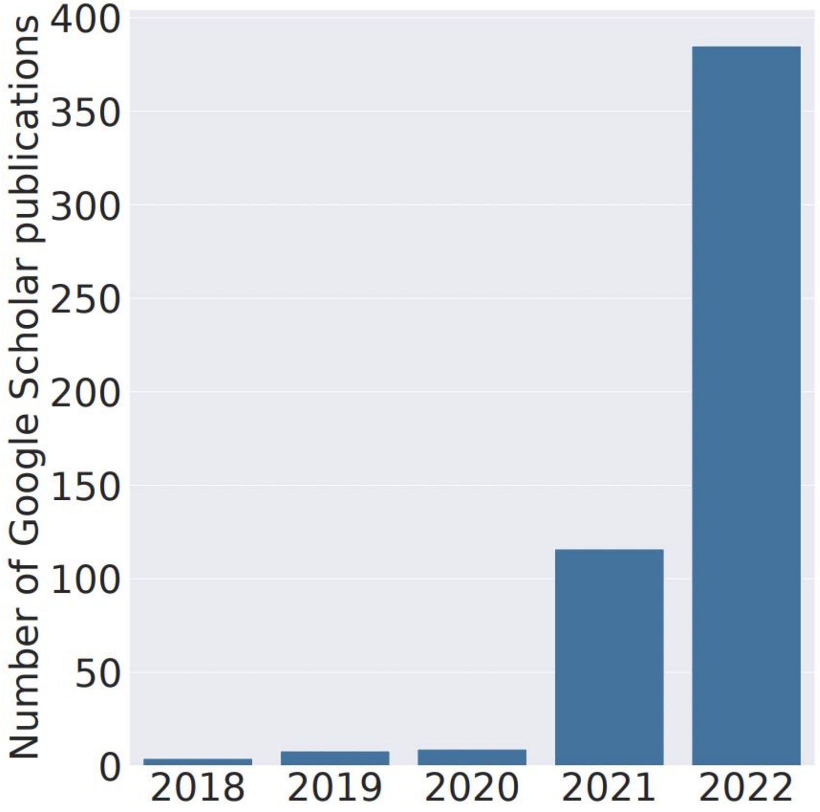
Trend: More Training Data for Better Models

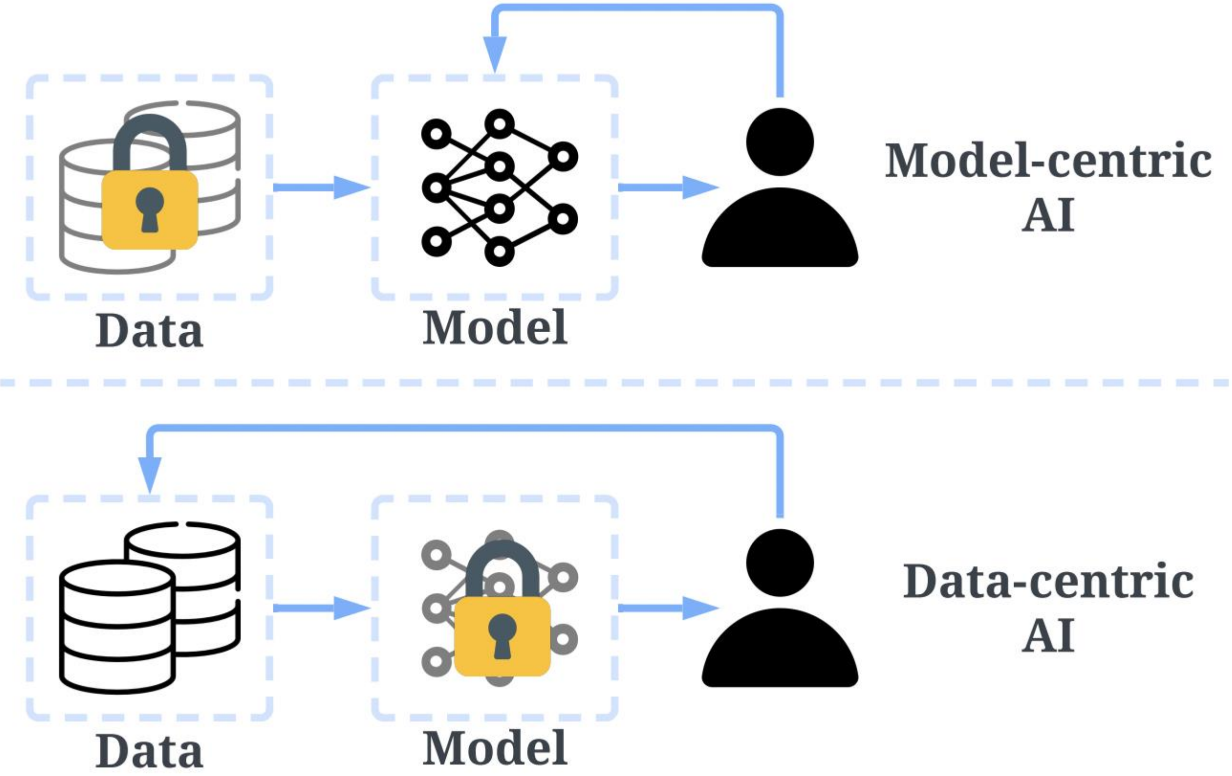


https://epoch.ai/data/notable-ai-models#explore-the-data

Shift from Model-Centric to Data-Centric AI

(exact match for "data-centric AI")





Greater focus on collecting data than designing models (e.g., GPT-1 to 4 series)

Zha et al. Data-centric AI: Perspectives and challenges. SDM 2023

How to Leverage Less Human Effort When Collecting Data?

Today's Topics

- Motivation
- Active learning
- Reinforcement learning
- Self-supervised learning
- And more…

How to more effectively leverage human supervision?

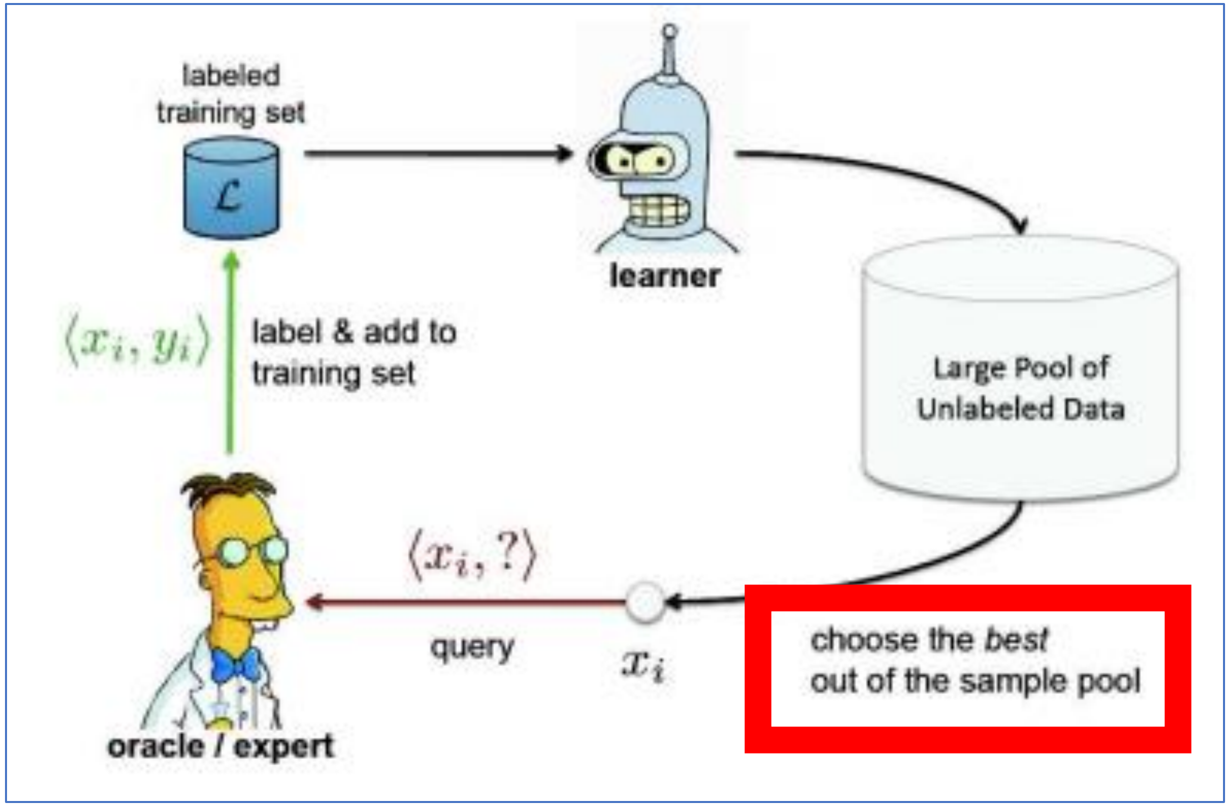
e.g., limited funding

e.g., limited access to
(expert) annotators

Active Learning: Keep Adding Labelled Training Examples for Most Informative Examples

Add more labeled training examples every *n* epochs based on what the refined model finds is hard

(different from curriculum learning because the added data needs labels to be collected)

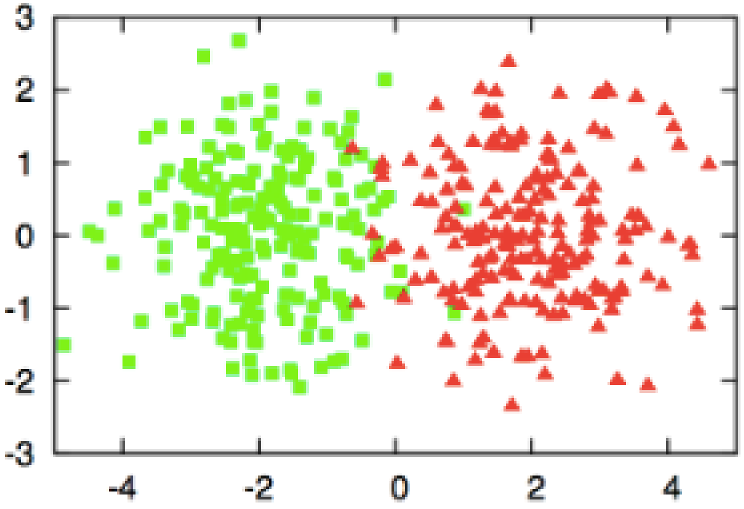


https://www.cs.utah.edu/~piyush/teaching/10-11-slides.pdf

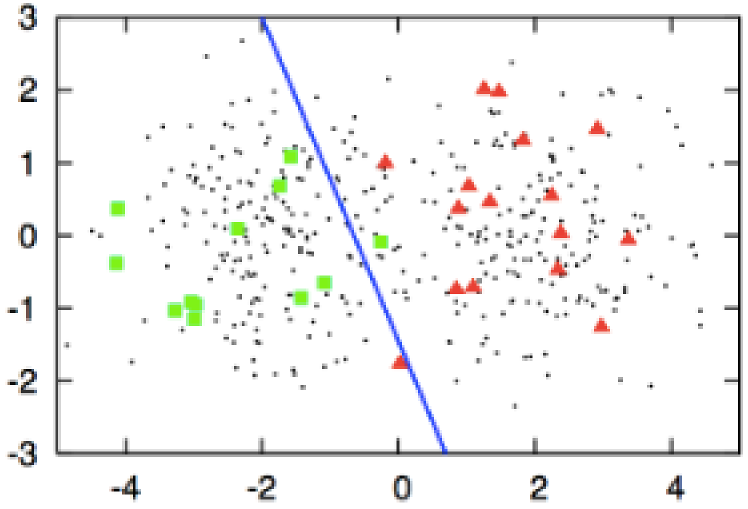
What approach might be effective in identifying the most informative data to label?

Uncertainty Sampling: Label Instance(s) Classifier is Most Uncertain About

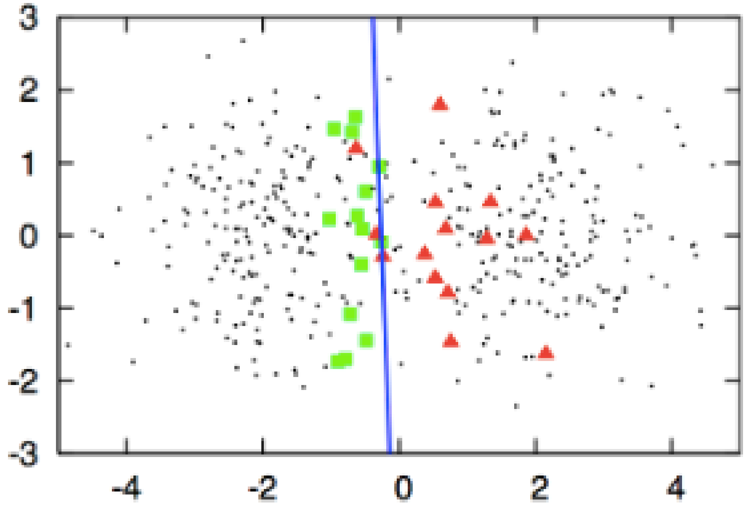
True Representation (Assume Labels Are Not Known)



Passive Learner (Random Selection)



Active Learner (Uncertainty Sampling)



http://burrsettles.com/pub/settles.activelearning.pdf

Uncertainty Estimation for Neural Networks Using Robustness Testing

Use model's predictions on random augmentations of the input to measure consistency/uncertainty; e.g.,

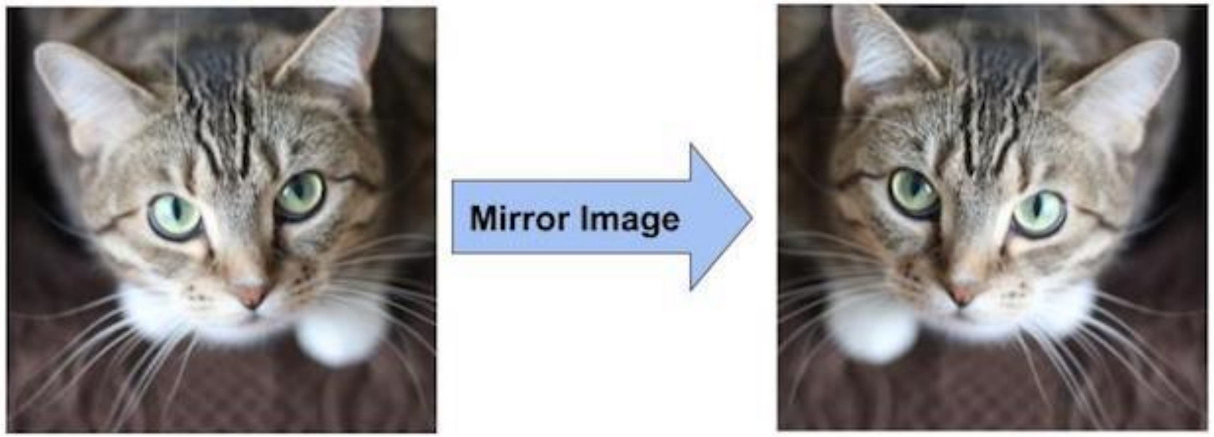
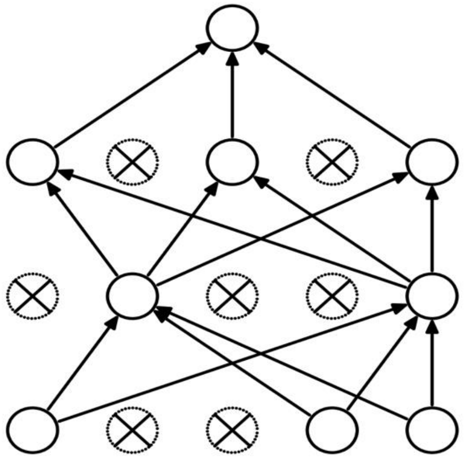


Figure Source: https://learnopencv.com/understanding-alexnet/

Elezi et al. Not all labels are equal: rationalizing the labeling costs for training object detection. CVPR 2022

Uncertainty Estimation for Neural Networks
Using Ensembles (Two Approaches)

1. Dropout with different masks at inference time


(b) After applying dropout.

2. Multiple neural networks

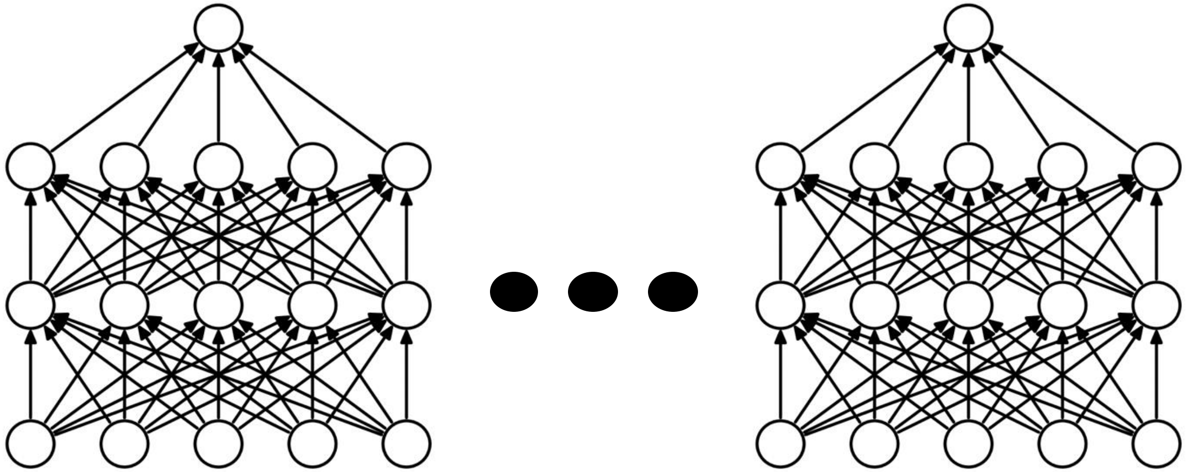


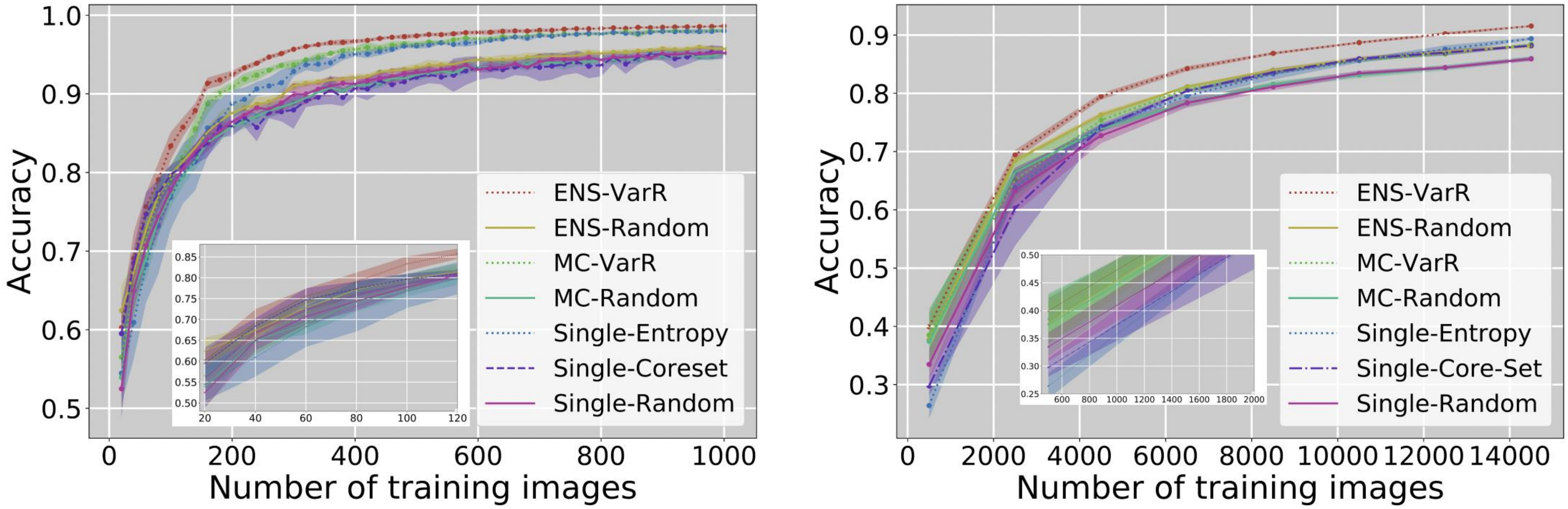
Figure Source: Srivastava et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research. 2014

Predicted softmax probabilities used to estimate uncertainty, with average taken across all ensemble's softmax distributions

Beluch et al. The power of ensembles for active learning in image classification. CVPR 2018

Uncertainty Estimation for Neural Networks
Using Ensembles (Two Approaches)

Active learning methods can lead to faster learning and reduced human annotation effort than passive (random) learning for two image classification datasets



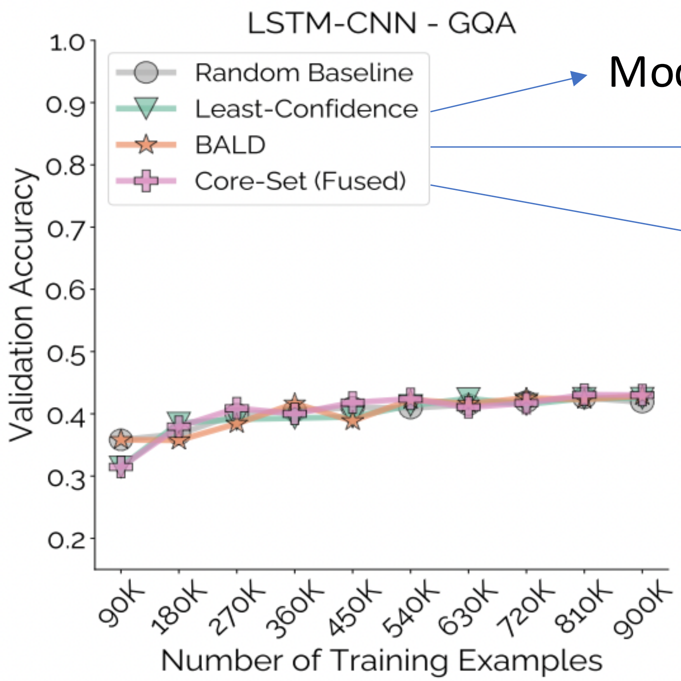
(a) MNIST on S-CNN

(b) CIFAR-10 on DenseNet

Beluch et al. The power of ensembles for active learning in image classification. CVPR 2018

Active Learning Techniques Have Mixed Results

- **Successes**: image classification, object detection
- **Failure**: VQA (e.g., AL methods label 10% of overall pool per iteration; initial model trained on 10% of pool)



Model's confidence in prediction

Based on ensemble from dropout

Examples capturing diversity in data pool

Karamcheti et al. Mind your outliers! Investigating the negative impact of outliers on active learning for visual question answering. ACL 2021

Active Learning Techniques Have Mixed Results

Why might AL methods perform comparable or worse to random selection?

- Challenging examples to learn are sampled; e.g.,

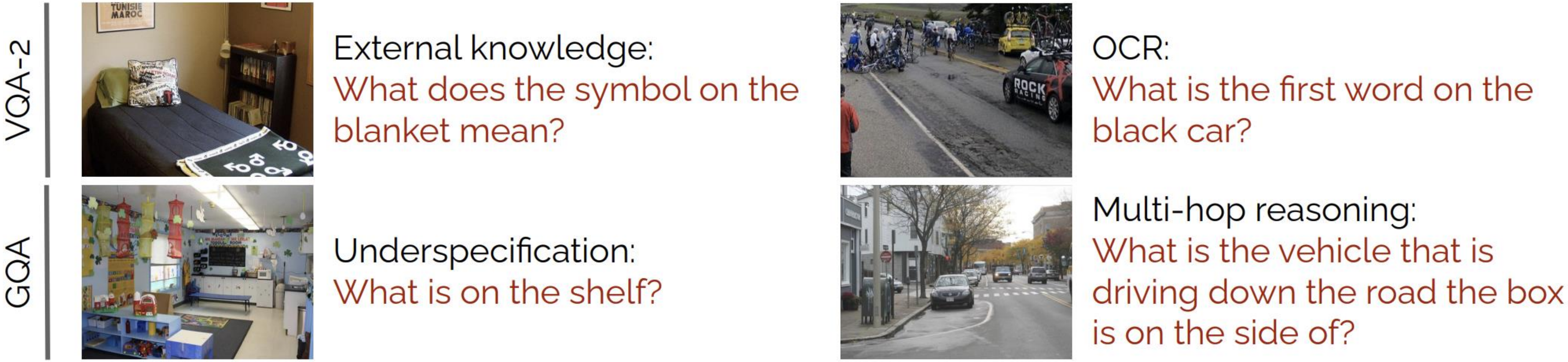
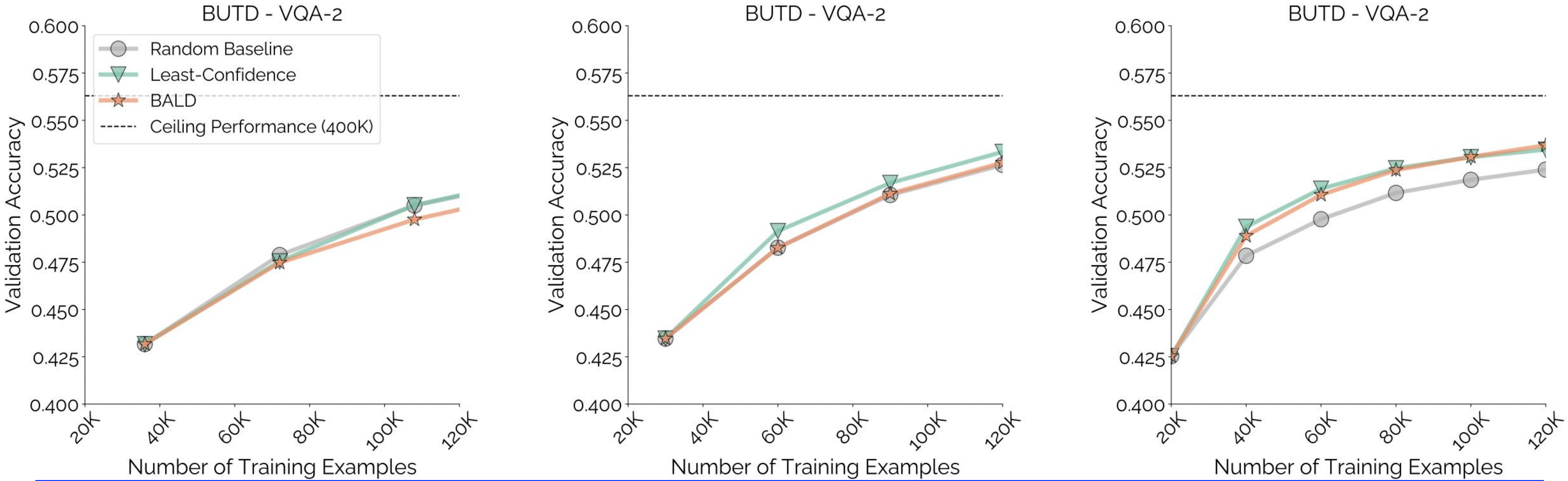


Figure 7: Example groups of collective outliers in the VQA-2 and GQA datasets.

Karamcheti et al. Mind your outliers! Investigating the negative impact of outliers on active learning for visual question answering. ACL 2021

Idea: Remove "Unlearnable" Data from Pool

Performance compared to random selection improves for AL approaches when removing "challenging" examples from data pool



(a) 10% of Dataset Removed (b) 25% of Dataset Removed (c) 50% of Dataset Removed

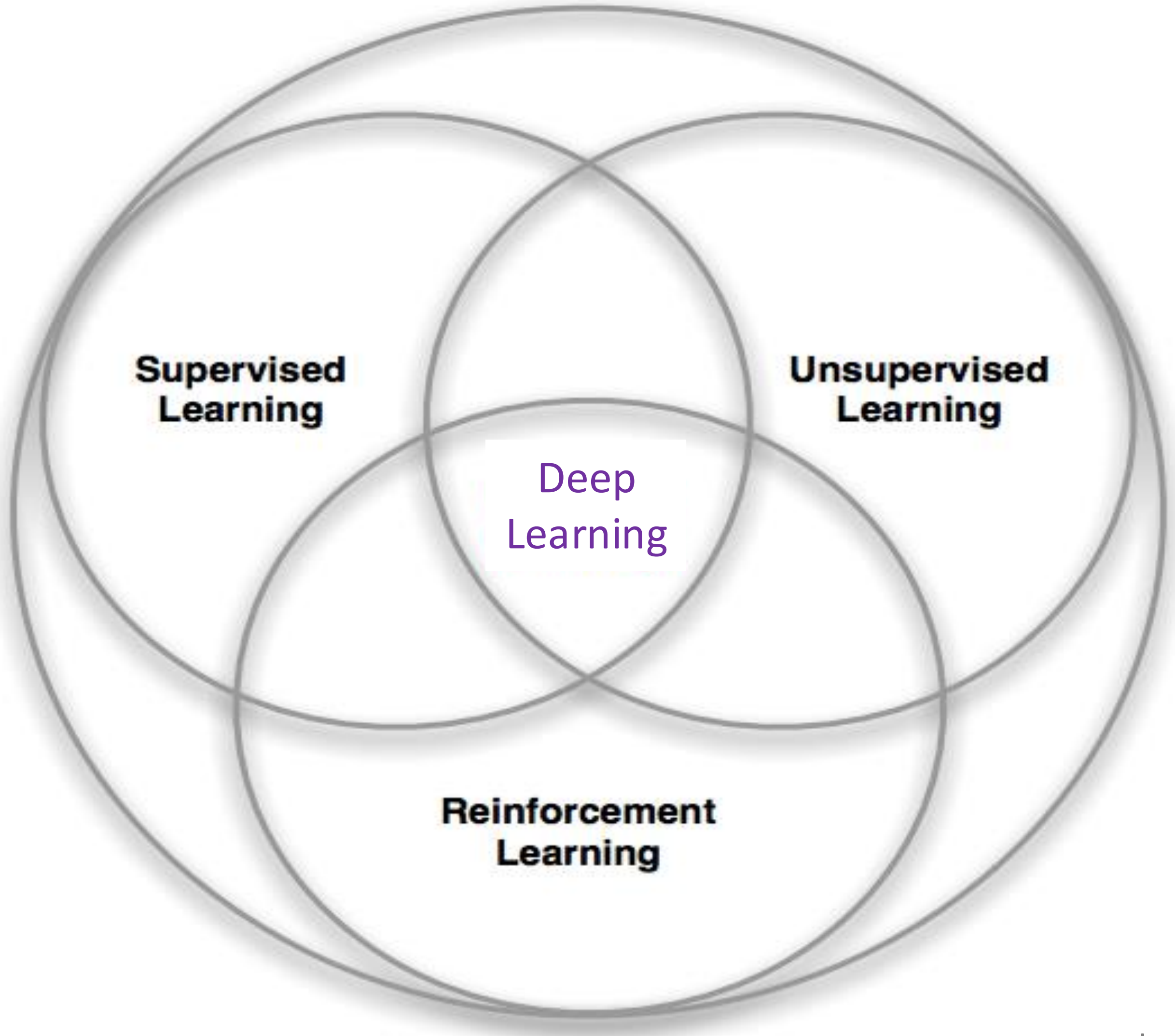
Karamcheti et al. Mind your outliers! Investigating the negative impact of outliers on active learning for visual question answering. ACL 2021

Today’s Topics

- Motivation
- Active learning
- Reinforcement learning
- Self-supervised learning
- And more…

How to provide coarser human supervision?

Reinforcement Learning Contextualized

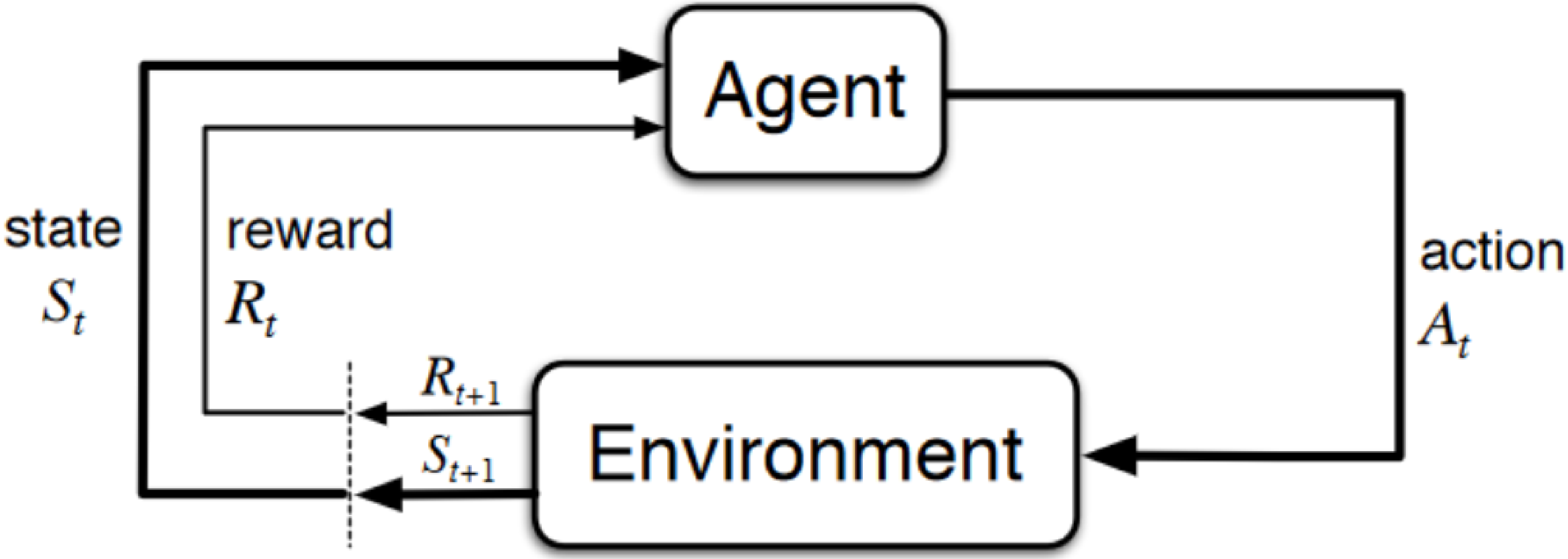


Unsupervised learning involves least amount of human effort followed by **reinforcement learning** with rewards and then **supervised learning** with labels

http://www0.cs.ucl.ac.uk/staff/d.silver/web/Teaching_files/intro_RL.pdf

Reinforcement Learning: Recall From Josh's Lecture On "Tuning Foundation Models"

Agent takes actions in an environment to maximize the total reward



https://towardsdatascience.com/applications-of-reinforcement-learning-in-real-world-1a94955bcd12

Application: Learning to Flip Pancakes

https://www.youtube.com/watch?v=W_gxLKSsSIE&list=PL5nBAYUyJTrM48dViibyi68urttMlUv7e

Application: Learning Dexterity

https://www.youtube.com/watch?v=jwSbzNHGflM

Theoretical Foundation of RL: Markov Decision Processes (MDP)

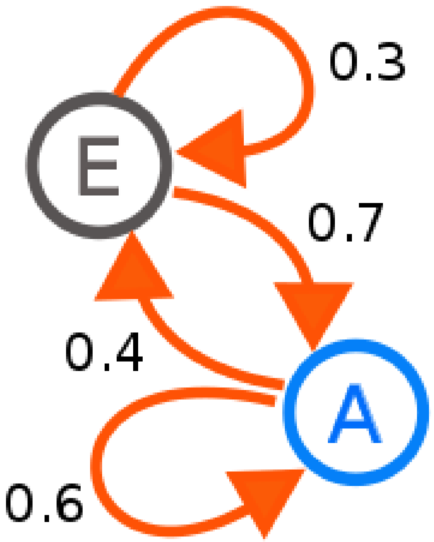
MDP consists of:

Markov process + Markov reward process + Actions

Markov Process (aka – Markov Chain)

- A set of states with transition probabilities defining system dynamics

e.g.,



  - How many states are in the above example?
- Markov property: only current state dictates future system dynamics

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

Theoretical Foundation of RL: Markov Decision Processes (MDP)

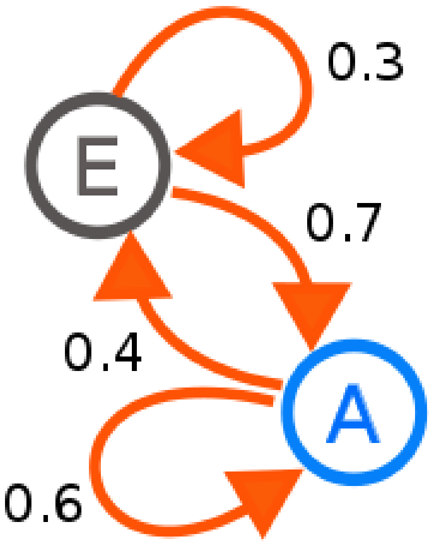
MDP consists of:

Markov process + $\boxed{\text{Markov reward process}}$ + Actions

Markov Reward Process

- Additional variable accumulated over time to reflect the reward

e.g.,



- A discount factor between 0 and 1, gamma, indicates how far in the future rewards are considered to estimate a state's expected reward

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

Theoretical Foundation of RL: Markov Decision Processes (MDP)

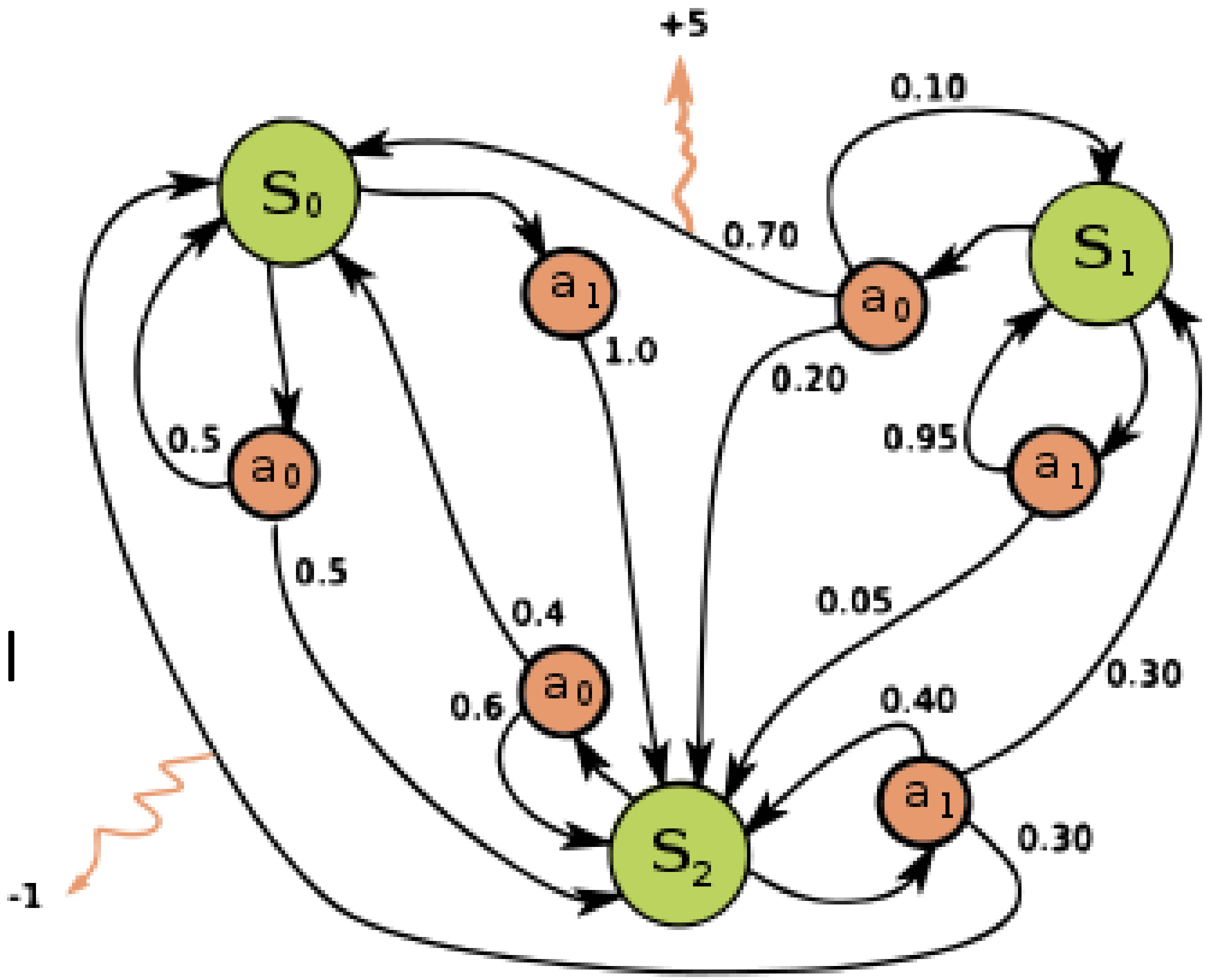
MDP consists of:

Markov process + Markov reward process + Actions

Third Ingredient of "Actions" Leads to a Markov Decision Process

- At each time step, the chosen action influences what will become the next state

- Probability allows for randomness (e.g., turn car wheel to go right on icy patch but car slips and continues straight)



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

Policy

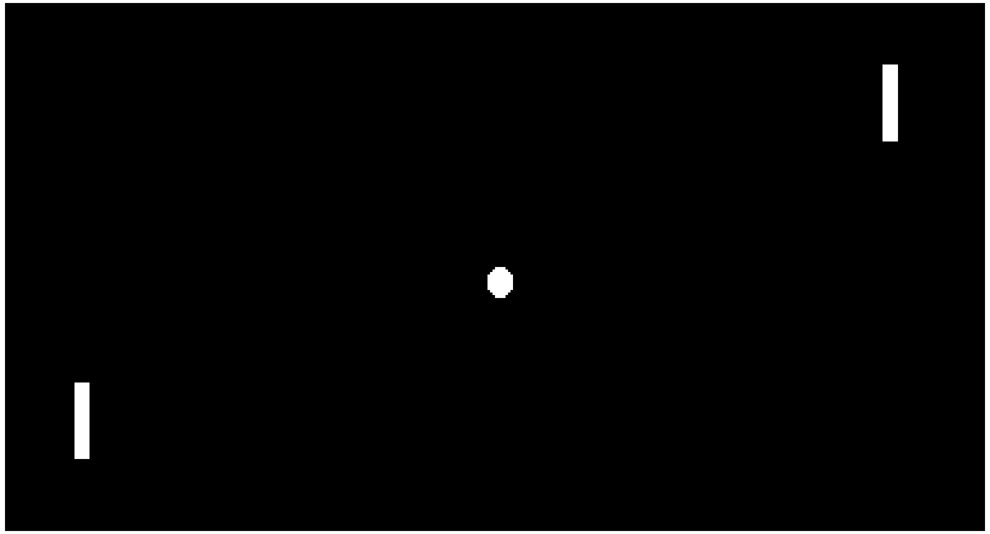
- Rules that dictate how an agent behaves
- Defined using a probability distribution over potential actions so there is randomness in the agent’s behavior
- RL goal: find a good policy

Basic Ingredients for RL Methods

1. Observations of environment
2. Possible actions
3. Rewards

Basic Ingredients for RL Methods; e.g., Pong

1. **Observations of environment**:

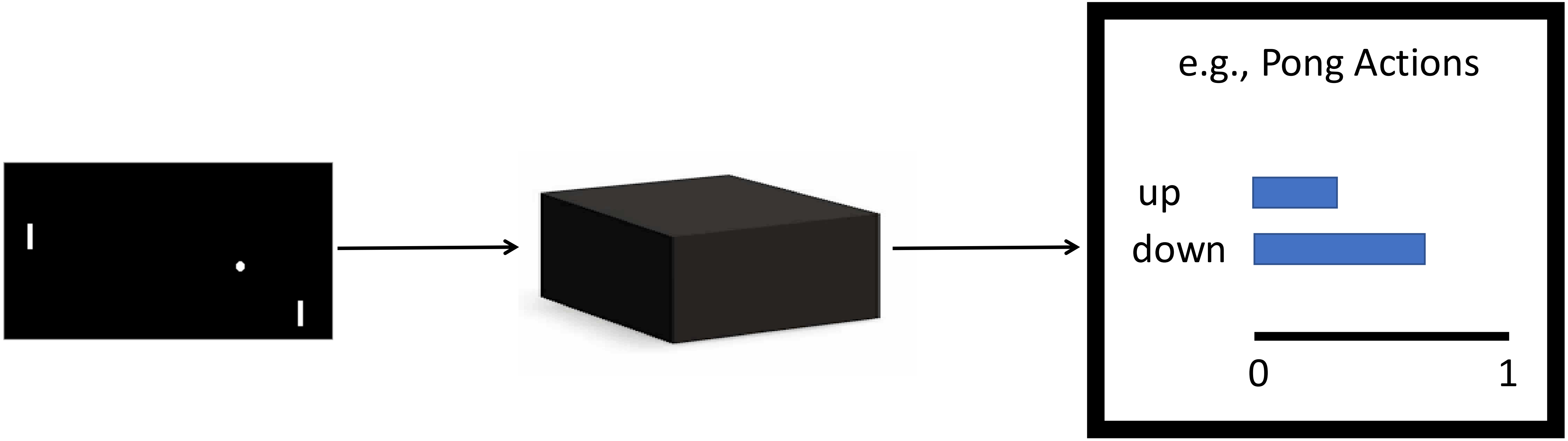


2. **Possible actions**: “up” and “down” paddle movements
3. **Rewards**: -1 if missed the ball; +1 reward if ball goes past opponent; 0 otherwise

Goal: Maximize rewards computing optimal “up” and “down” paddle movements

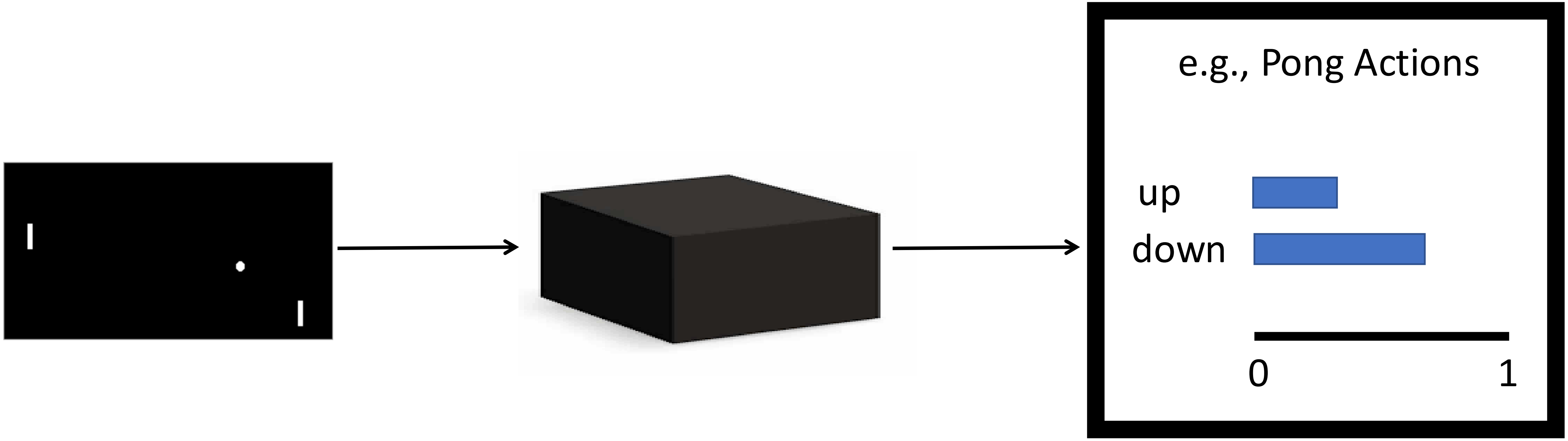
http://karpathy.github.io/2016/05/31/rl/

Policy Gradients: Approach



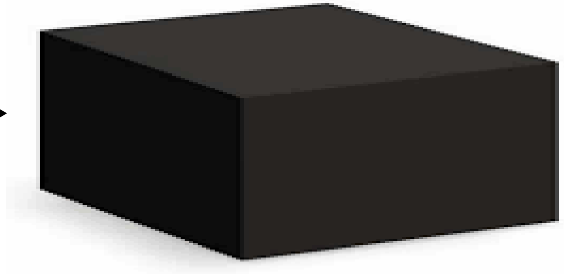
Policies (i.e., rules dictating how an agent behaves) are represented using a probability for each possible action

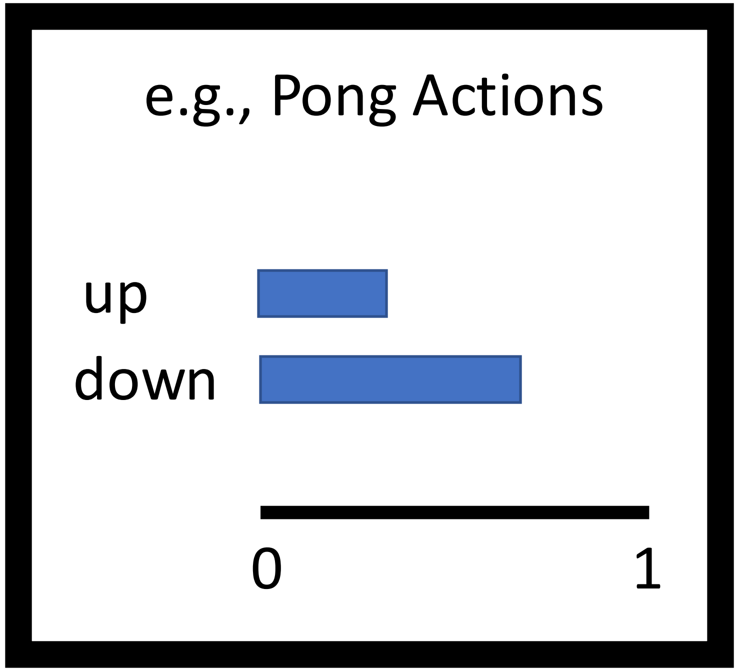
Policy Gradients: Approach



Neural network trained to increase probability of actions leading to a good total reward and decrease probability of actions leading to a bad total reward

Policy Gradients: Approach

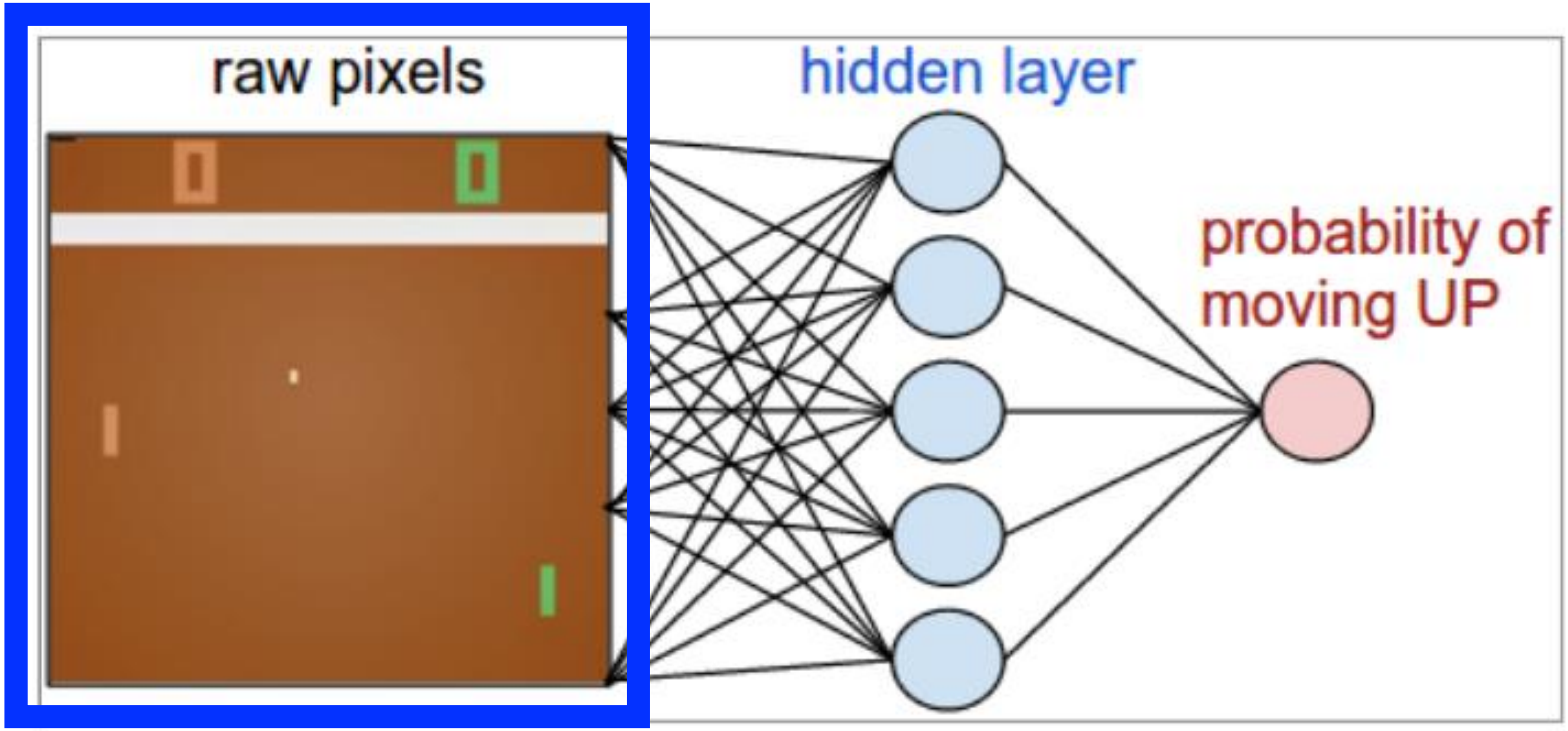




How does this approach support "exploration"?

e.g., Learning Pong
(2-layer NN with 200 hidden units)

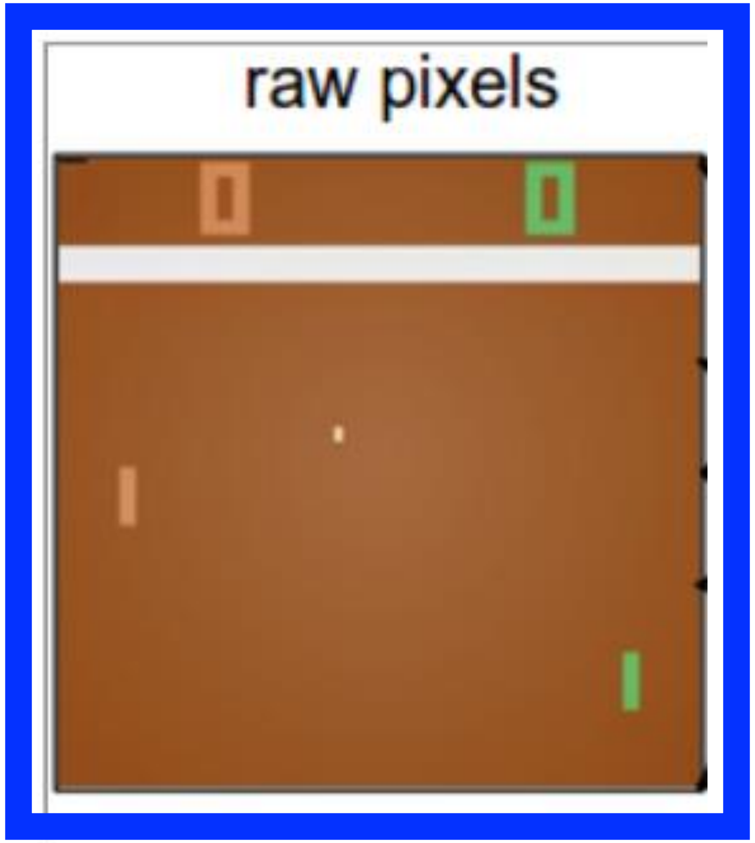
Given game state (as image), decide if to move paddle up (vs down)



http://karpathy.github.io/2016/05/31/rl/

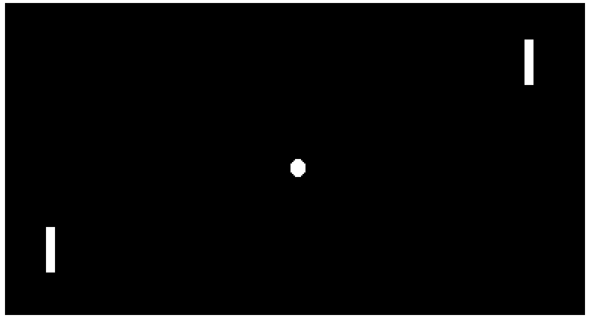
e.g., Learning Pong
(2-layer NN with 200 hidden units)

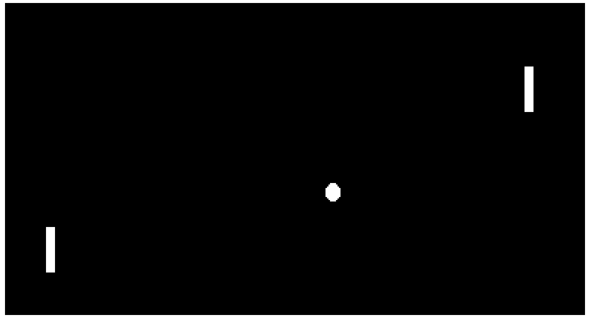
How to capture motion in game state (i.e., image)?



Use difference image

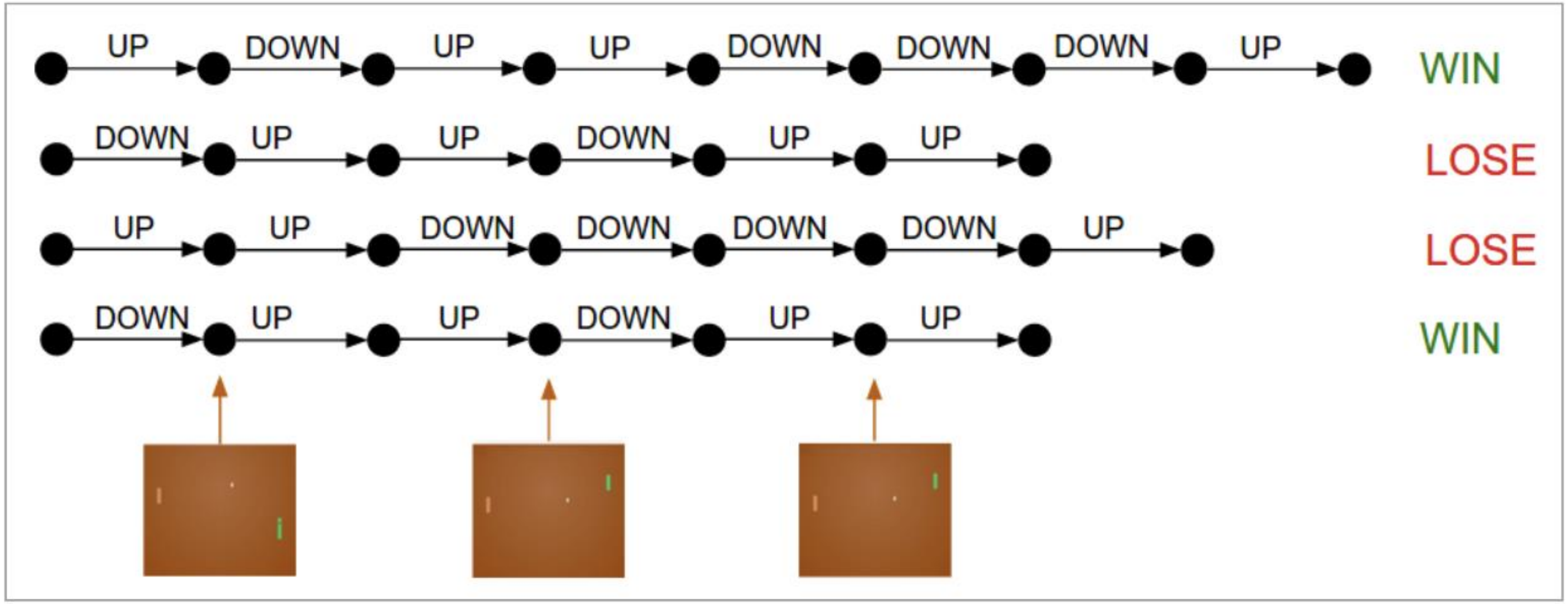
(i.e., subtract frame current from last frame)





http://karpathy.github.io/2016/05/31/rl/

e.g., Learning Pong: Training Protocol



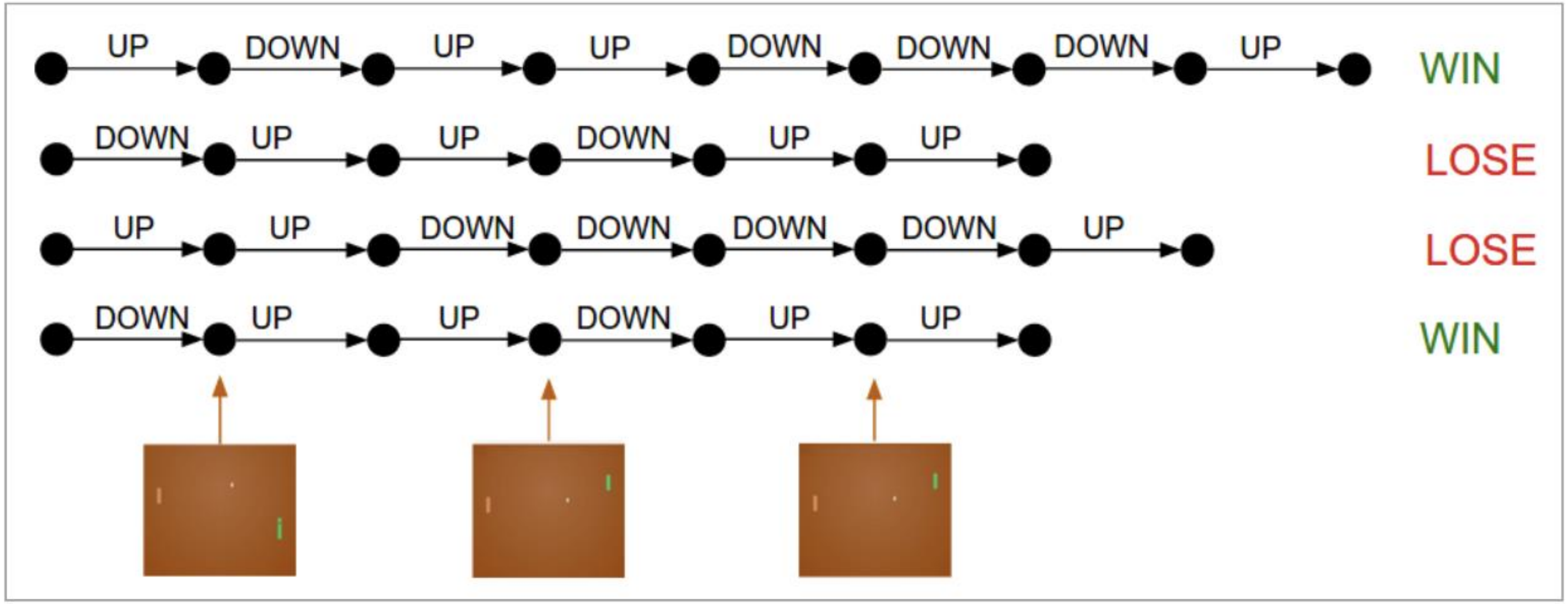
Assume 100 games played with 200 images/game

How many (action) decisions were made?

- 100x199

http://karpathy.github.io/2016/05/31/rl/

e.g., Learning Pong: Training Protocol

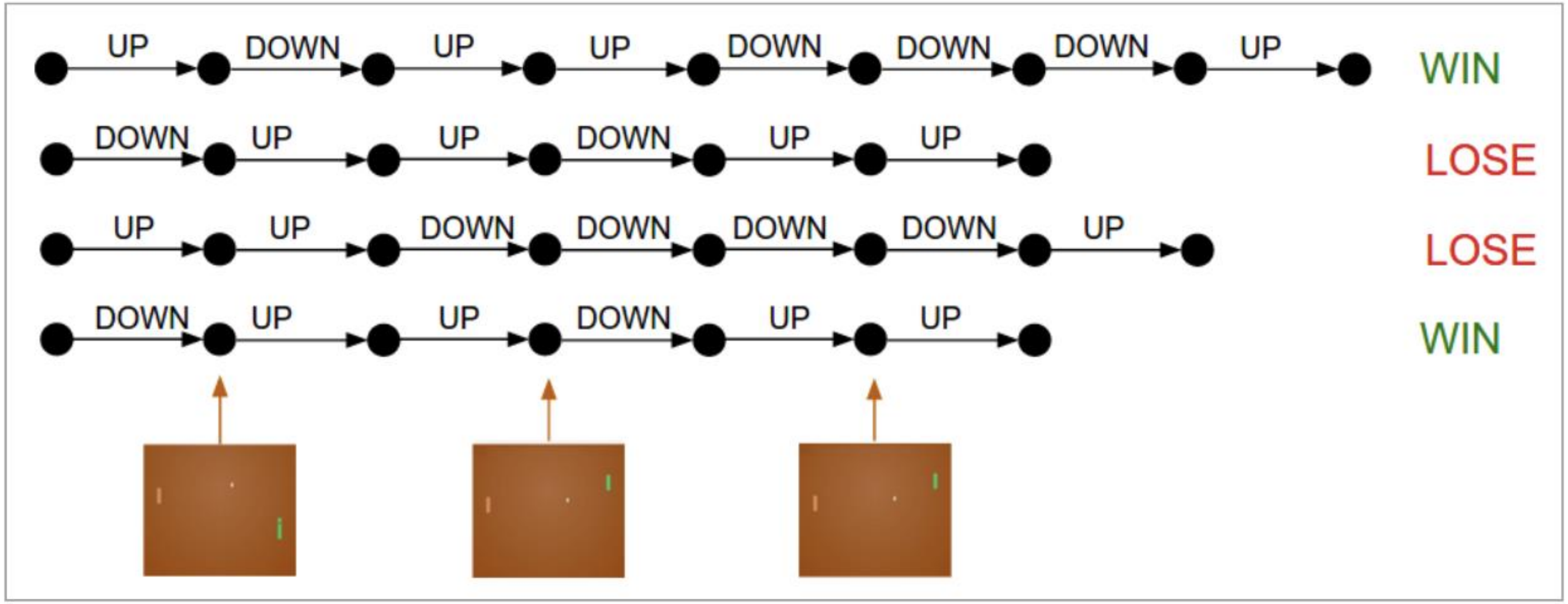


Assume 100 games played with 200 images/game; 12 games won & 88 lost

- How many winning decisions were made?
  - 2,388 (i.e., 12 x 199)
- How many losing decisions were made?
  - 17,512 (i.e., 88 x 199)

http://karpathy.github.io/2016/05/31/rl/

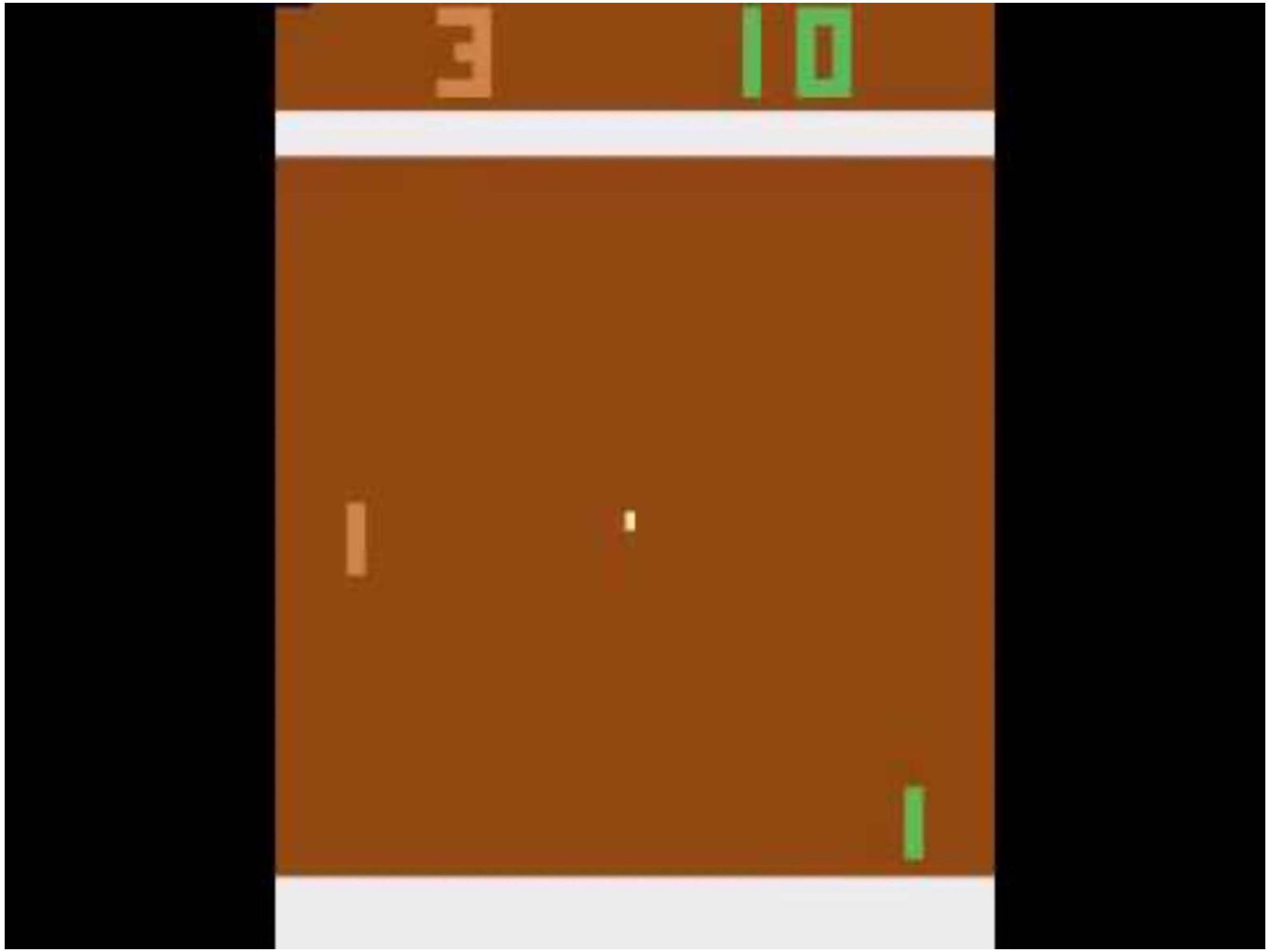
e.g., Learning Pong: Training Protocol



After 100 games, gradient updated to encourage actions that led to good outcomes (i.e., 2,388 winning up/downs) and discourage actions that led to bad outcomes (17,512 losing up/downs)

http://karpathy.github.io/2016/05/31/rl/

e.g., Pong Model: RL Model vs Pong’s AI Model



https://www.youtube.com/watch?v=YOW8m2YGtRg&t=16s

Why Reinforcement Learning is Difficult

- Agent must infer what it did wrong/right and so how performance can be maintained/improved based on (delayed) rewards; e.g., win/lose pong

- Agent needs to strike the appropriate balance between exploration and exploitation; e.g., order one's favorite food vs something new

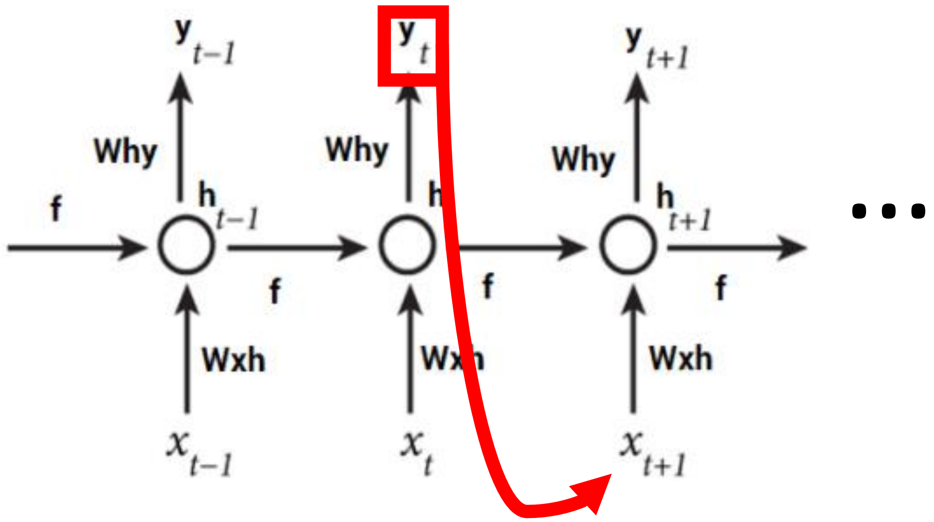
Today's Topics

- Motivation
- Active learning
- Reinforcement learning
- Self-supervised learning
- And more…

How to avoid extra human supervision?

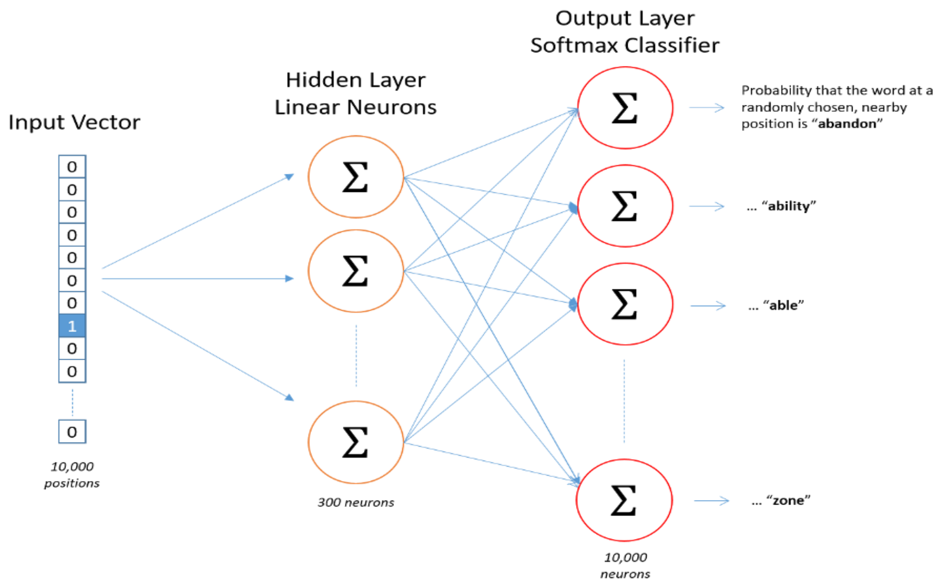
Recall, self-supervised learning in previous lectures; e.g.,

RNNs
(e.g., predict next character)



https://www.analyticsvidhya.com/blog/2017/12/introduction-to-recurrent-neural-networks/

Word embeddings
(e.g., predict nearby word for given word for word2vec)



https://towardsdatascience.com/word2vec-skip-gram-model-part-1-intuition-78614e4d6e0b

Self-Supervised Learning: Data Is Supervision

- Relatively Cheap
- Can Collect Data Fast

https://lovevery.com/community/blog/child-development/the-surprising-learning-power-of-a-household-mirror/

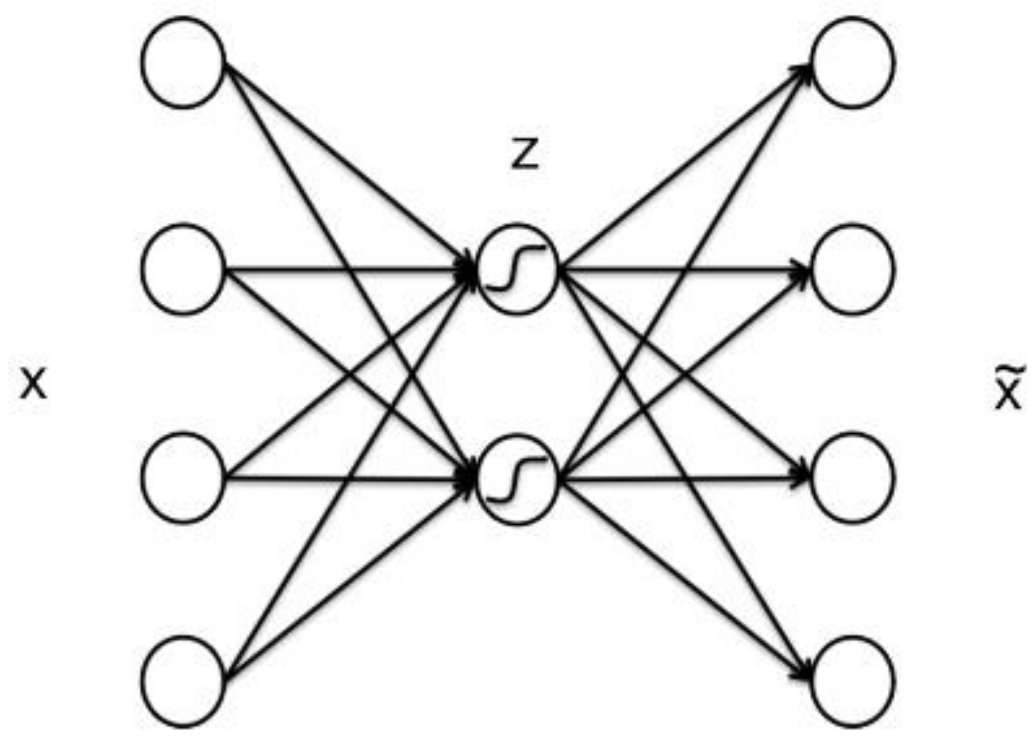
https://www.rockettes.com/blog/how-to-use-the-mirror-in-dance-class/

Types of Approaches

- Generative-based methods
- Generative adversarial networks
- Context-based methods

e.g., Image Autoencoder

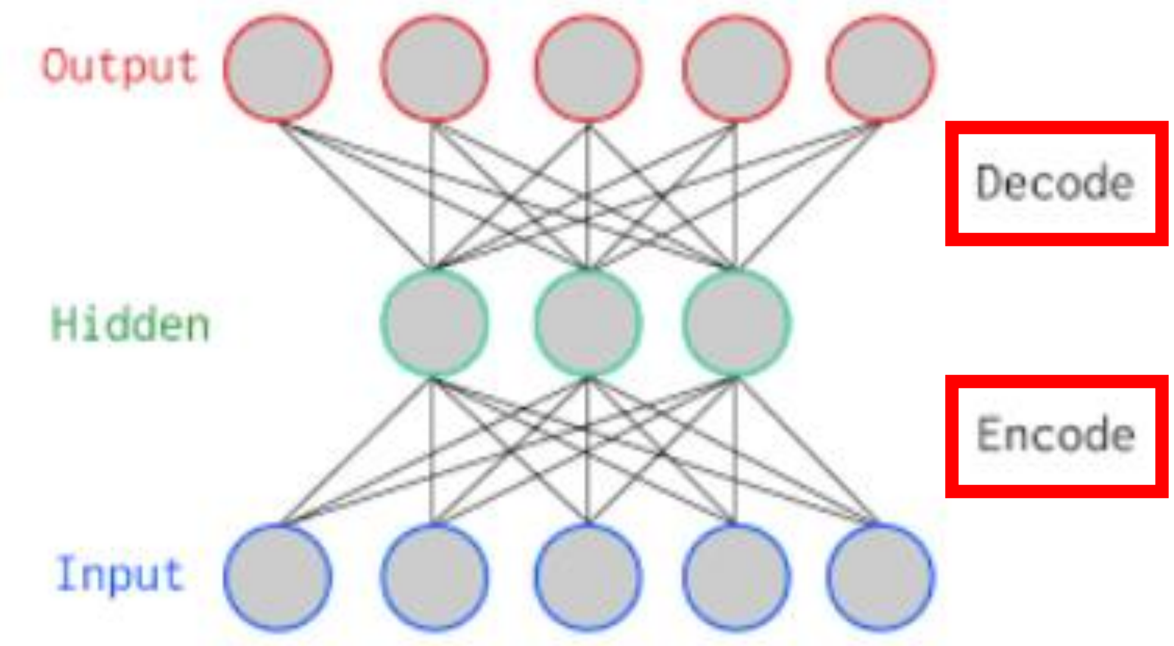
- Learn to copy the input to the output



https://lazyprogrammer.me/a-tutorial-on-autoencoders/

e.g., Image Autoencoder Architecture

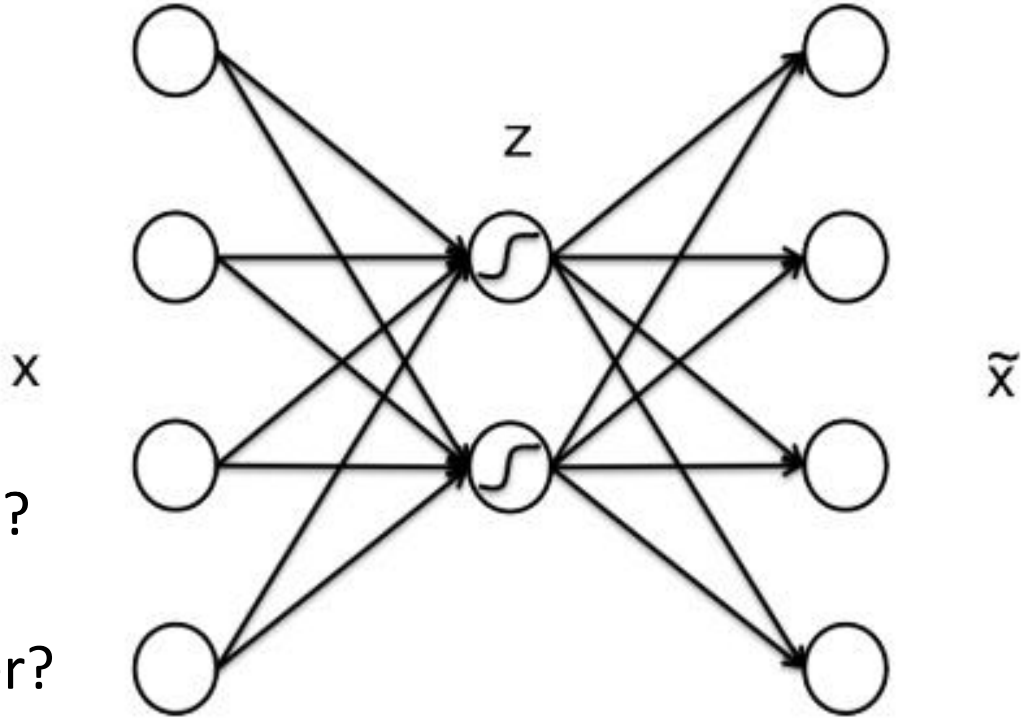
- Consists of two parts:
  - **Encoder**: compresses inputs to an internal representation
  - **Decoder**: tries to reconstruct the input from the internal representation



https://www.datacamp.com/community/tutorials/autoencoder-keras-tutorial

e.g., Image Autoencoder Architecture

- Given this input 620 x 426 image (264,120 pixels):

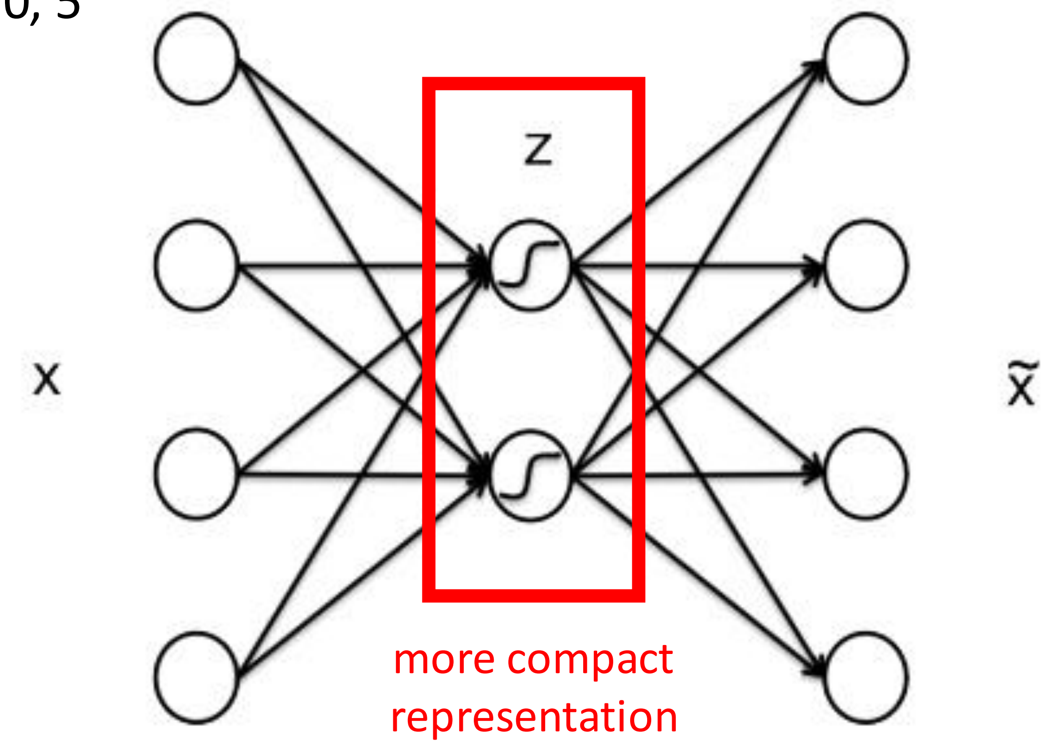


  - What would a perfect autoencoder predict?
    - Itself
  - What number of nodes are in the final layer?
    - 264,120

https://lazyprogrammer.me/a-tutorial-on-autoencoders/

e.g., Image Autoencoder Idea

- Intuition: which number sequence is easier to remember?
  - **A:** 30, 27, 22, 11, 6, 8, 7, 2
  - **B:** 30, 15, 46, 23, 70, 35, 106, 53, 160, 80, 40, 20, 10, 5
- **B:** need learn only two rules
  - If even, divide by 2
  - If odd, multiply by 3 and add 1



https://lazyprogrammer.me/a-tutorial-on-autoencoders/

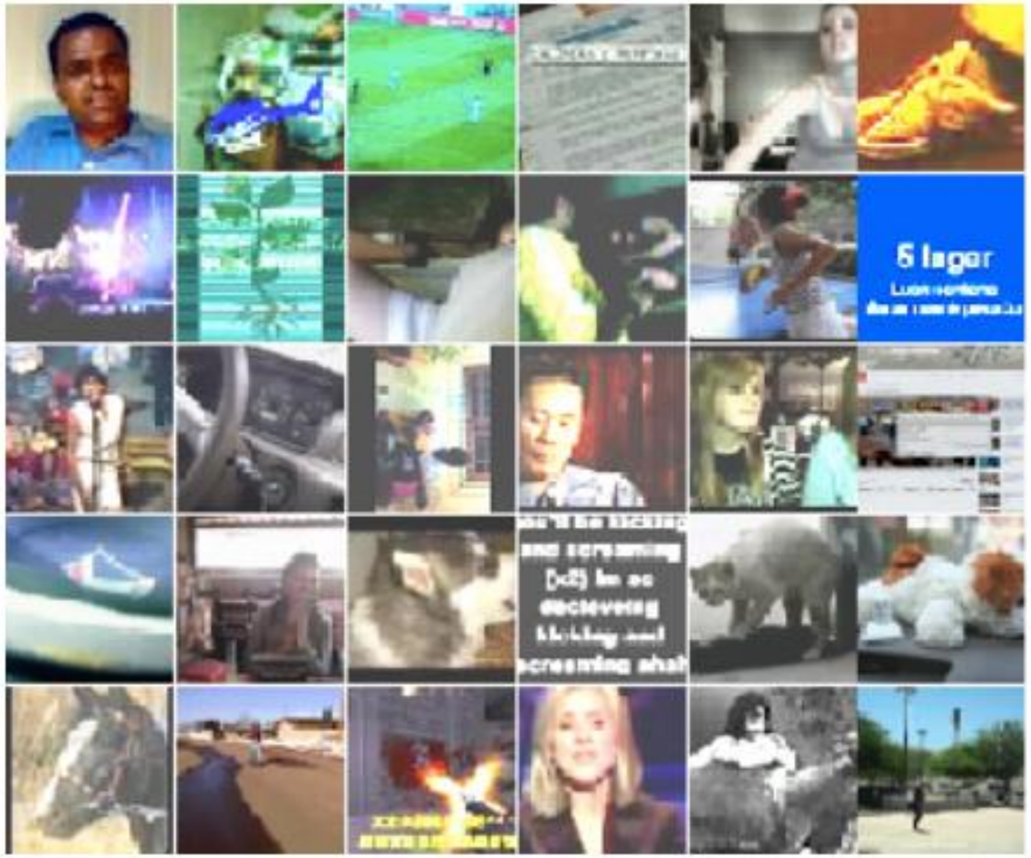
e.g., Image Autoencoder Training

Repeat until stopping criterion met:

1. **Forward pass**: propagate training data through network to make prediction
2. **Backward pass**: using predicted output, calculate error gradients backward
3. Update each weight using calculated gradients

e.g., Image Autoencoder Features

- e.g., training data:
  - 1 image taken from 10 million YouTube videos
  - Each image is in color and 200x200 pixels

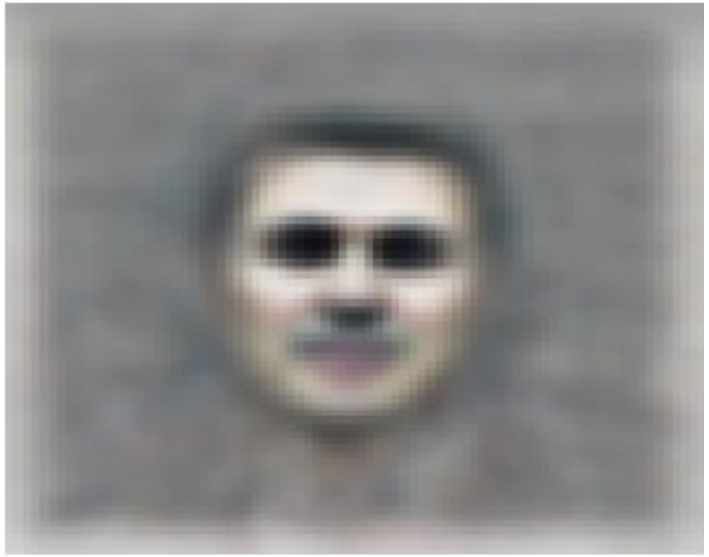


- What features do you think it learned?

Quoc V. Le et al., Building High-level Features Using Large Scale Unsupervised Learning; ICML 2013.

e.g., Image Autoencoder Features

- Features learned include:



human face

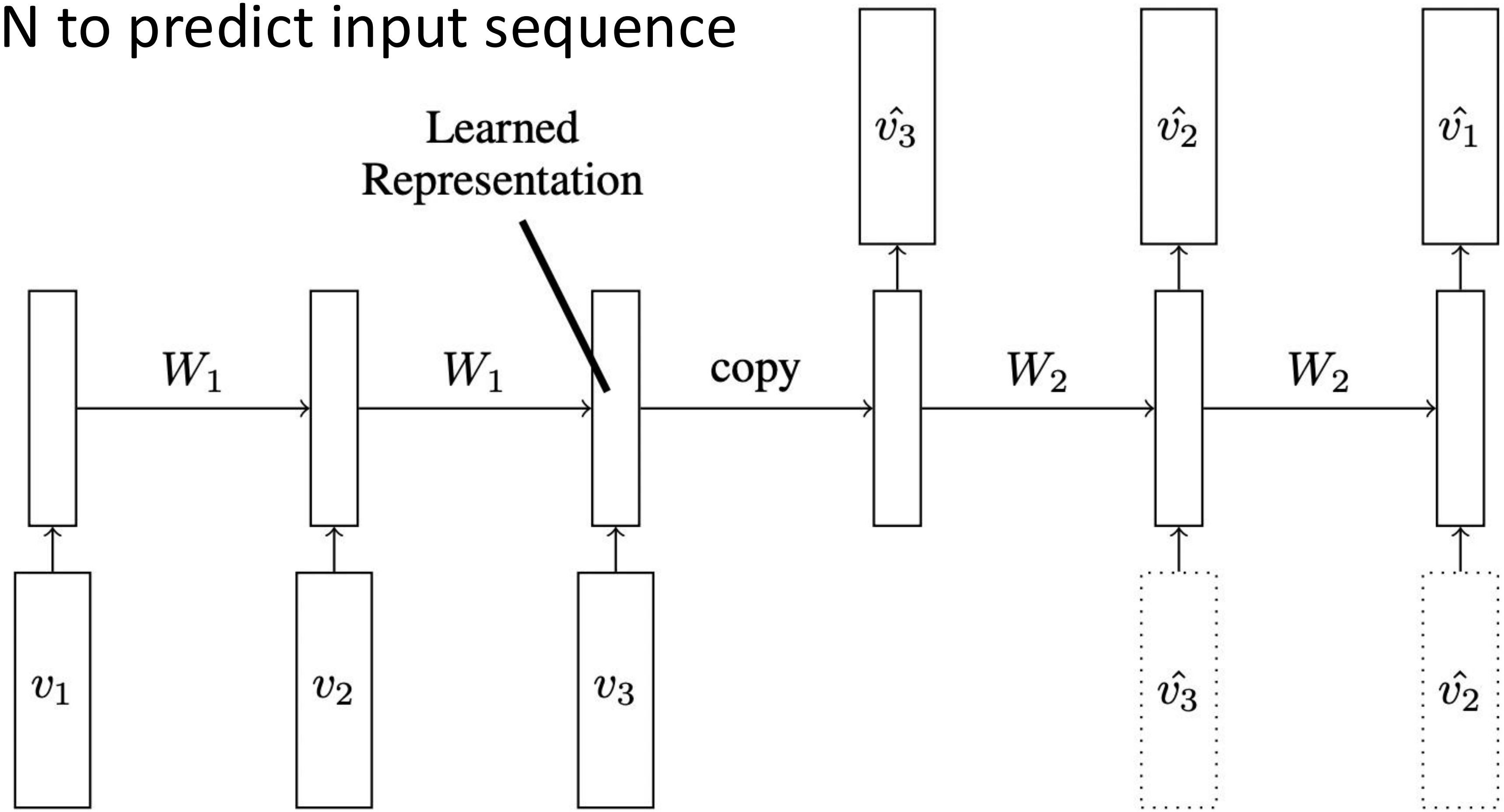
cat face

human body

Quoc V. Le et al., Building High-level Features Using Large Scale Unsupervised Learning; ICML 2013.

e.g., Video Autoencoder

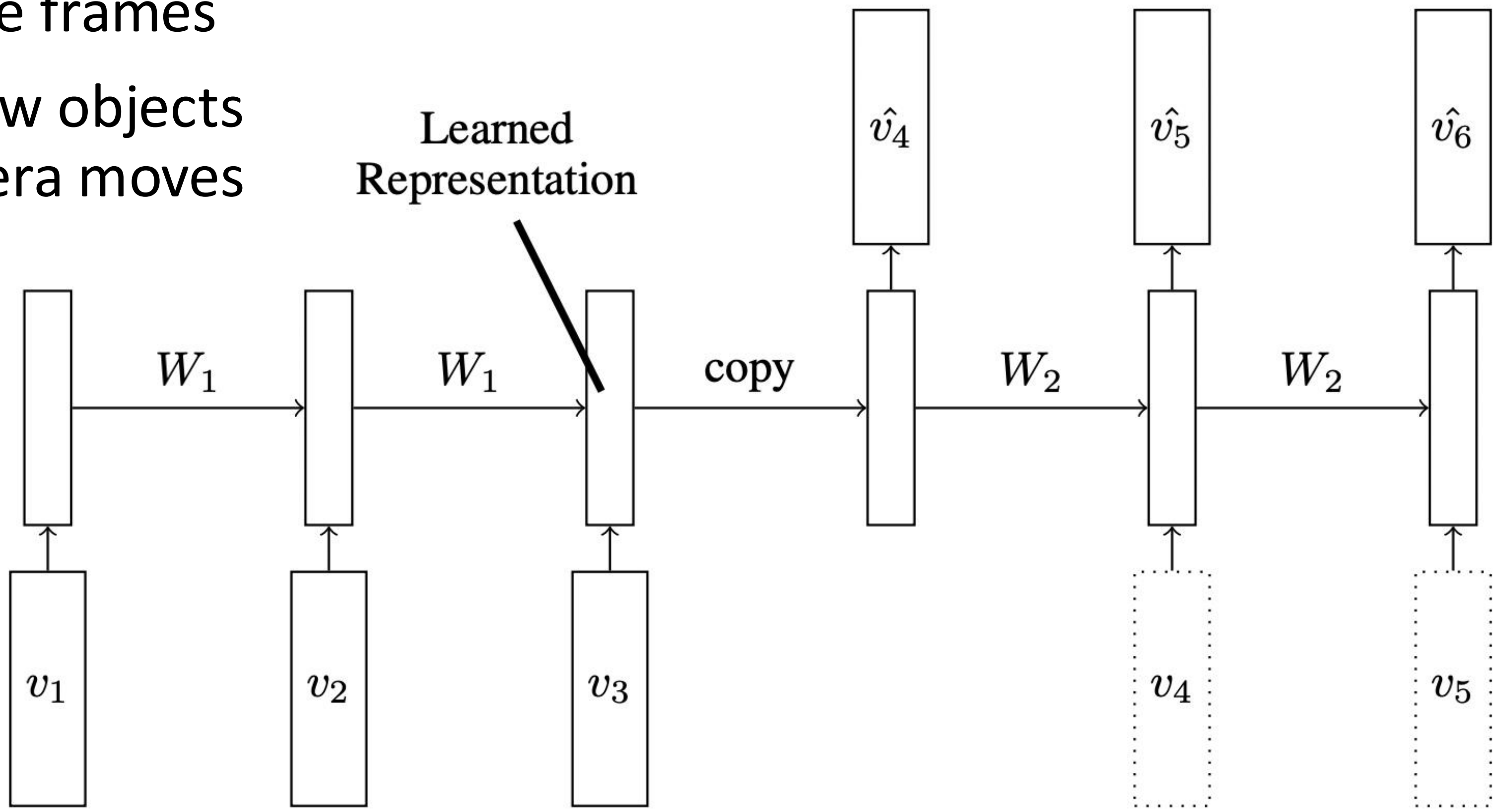
- Train RNN to predict input sequence



Srivastava et al., Unsupervised Learning of Video Representations using LSTMs; ICML 2015.

e.g., Video Prediction

- Train RNN to predict future frames
- Limitations: identifying new objects and background as a camera moves



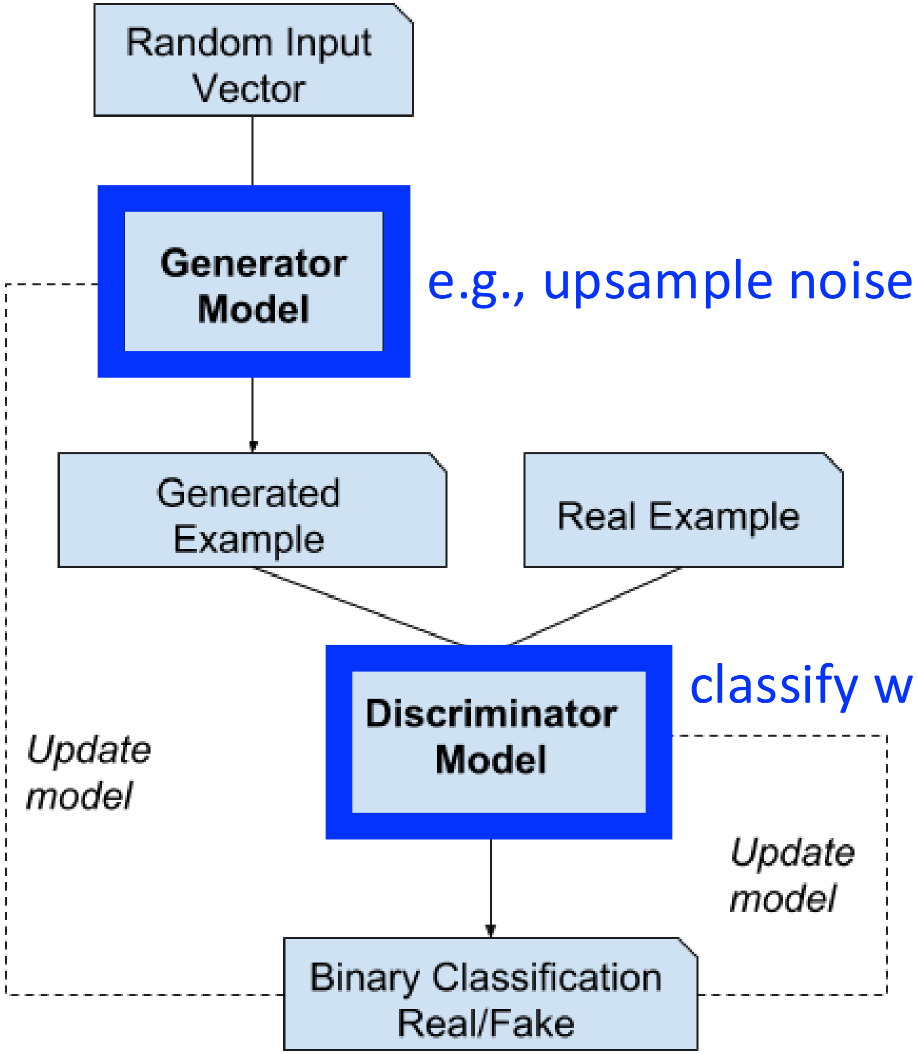
What type of features might be learned?

Srivastava et al., Unsupervised Learning of Video Representations using LSTMs; ICML 2015.

Types of Approaches

- Generative-based methods
- Generative adversarial networks
- Context-based methods

GAN: Basic Architecture



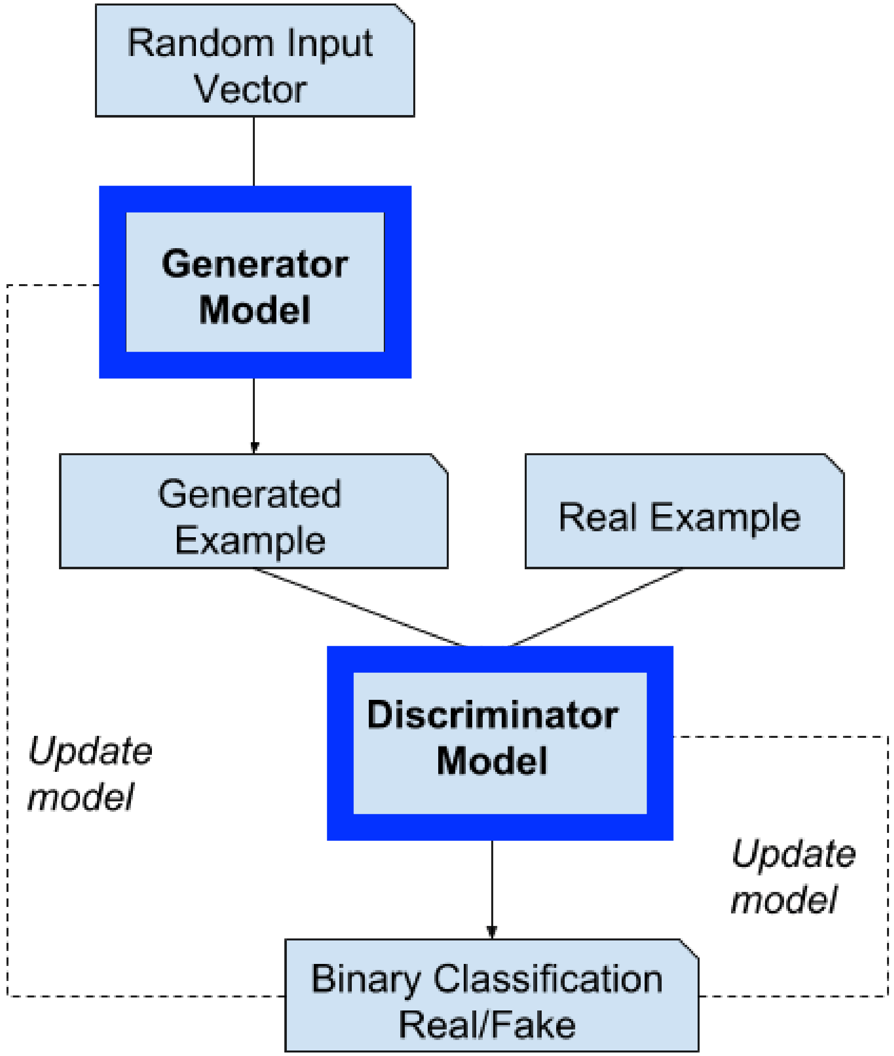
e.g., upsample noise to image of desired size

classify which examples are real

Consists of two models that compete against each other

https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/

GAN: Training



The two models are iteratively trained separately

- Train discriminator using fake and real images
- Train generator using just fake images and penalize it when the discriminator recognizes images are fake

https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/

GAN: Discriminator Loss Function

Discriminator tries to minimize classification error

Discriminator wants a value of 1 for real images

Discriminator wants a value of 0 for fake images

$$J^{(D)} = -\frac{1}{2}\mathbb{E}_{\boxed{\boldsymbol{x}\sim p_{\text{data}}}} \log \boxed{D(\boldsymbol{x})} - \frac{1}{2}\mathbb{E}_{\boxed{\boldsymbol{z}}} \log\left(1 - \boxed{D\left(G(\boldsymbol{z})\right)}\right)$$

Real image

Input noise

GAN: Generator Loss Function

Generator tries to maximize classification error

$$J^{(G)} = -J^{(D)}$$

Want the discriminator to mistakenly arrive at a value of 1 for fake images

$$J^{(G)} = -\frac{1}{2}\mathbb{E}_{\boxed{\boldsymbol{z}}} \log \boxed{D\left(G(\boldsymbol{z})\right)}$$

Input noise

https://arxiv.org/pdf/1701.00160.pdf

DGANs: GANs that Use Convolutional Layers

Bedrooms generated by observing over 3M bedroom images

Radford et al. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv 2015.

DGANs: GANs that Use Convolutional Layers

What objects does it learn to generate?

Radford et al. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv 2015.

DGANs: GANs that Use Convolutional Layers

What objects may it not have learned to generate?

Radford et al. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv 2015.

DGANs: GANs that Use Convolutional Layers

Faces generated by observing over 3M images of 10K people

Radford et al. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv 2015.

DGANs: GANs that Use Convolutional Layers

What does it generate poorly or not all?

Radford et al. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv 2015.

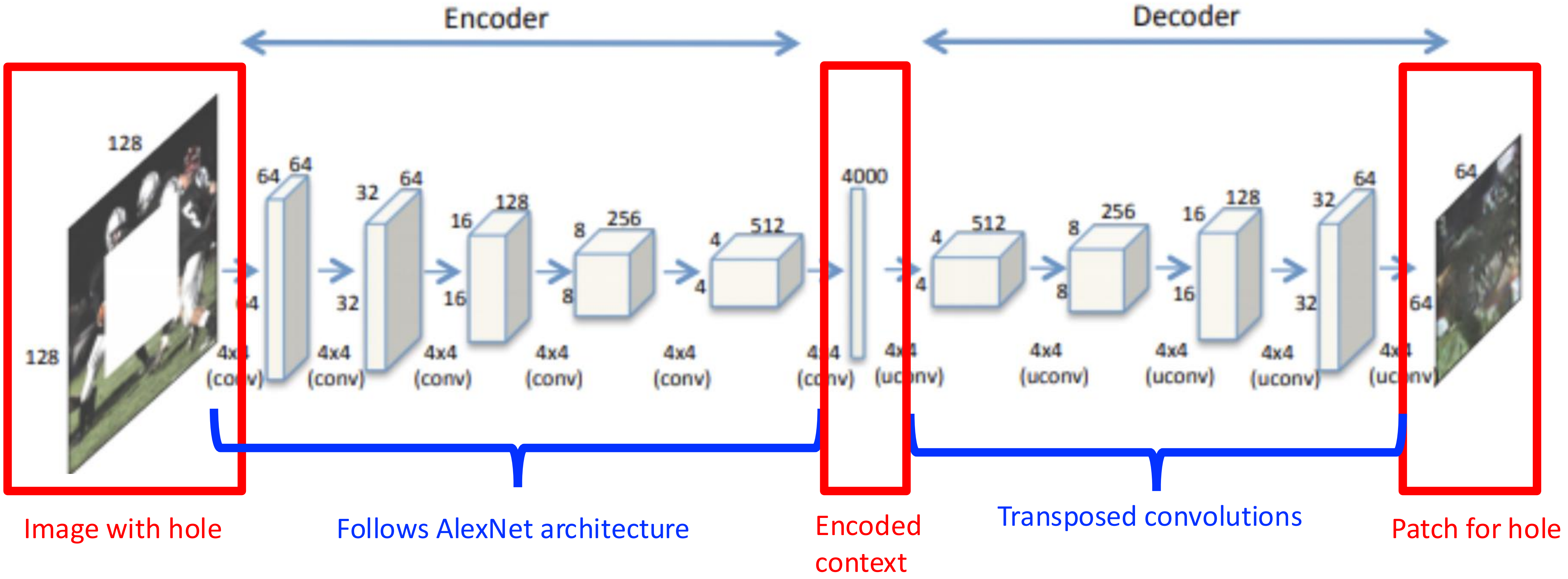
Another Task: Hole Filling

- What might fit into this hole?

- Many items may plausibly fit into the hole:

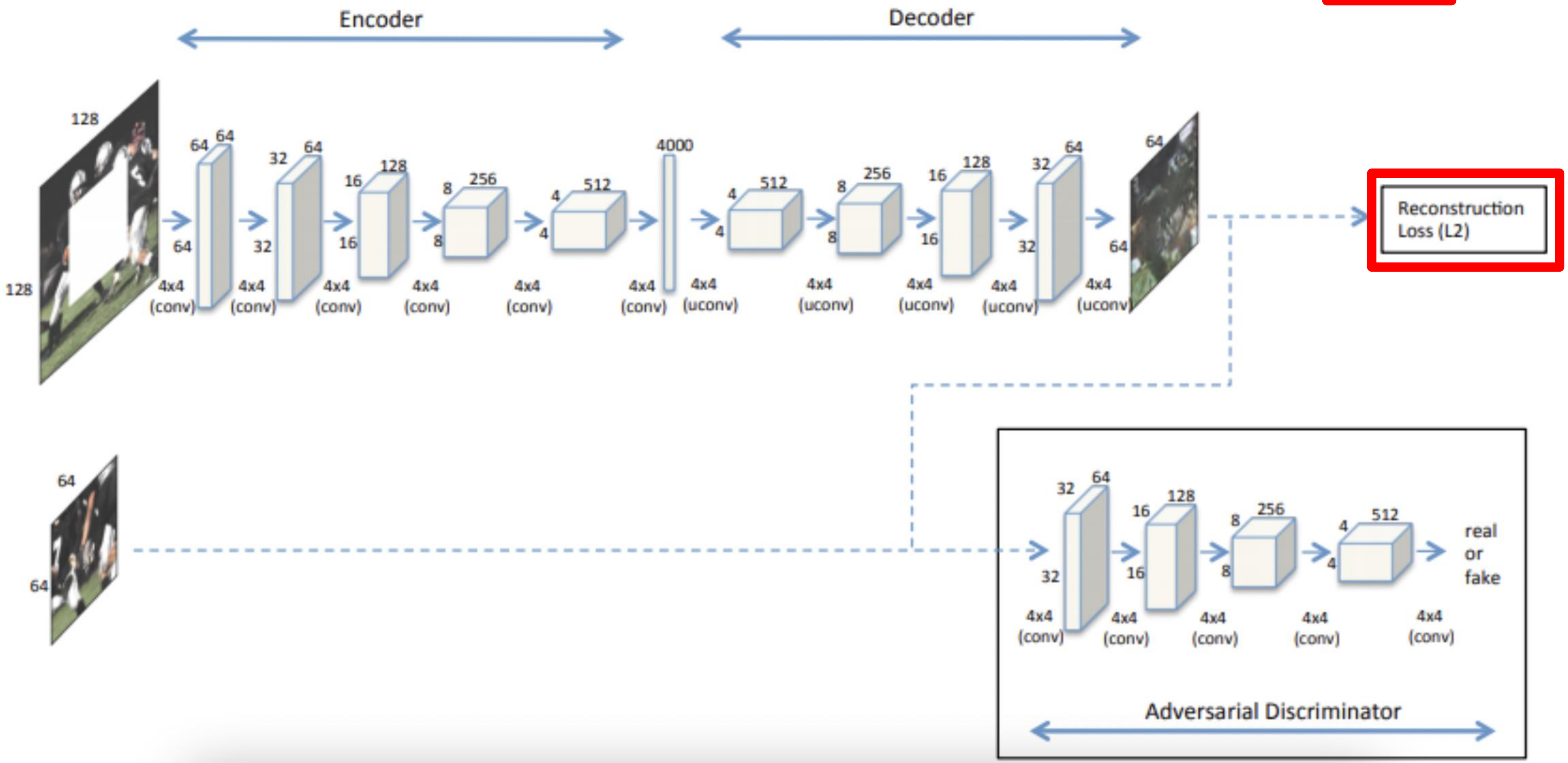
- Challenge: have up to 1 known ground truth region per hole

Architecture



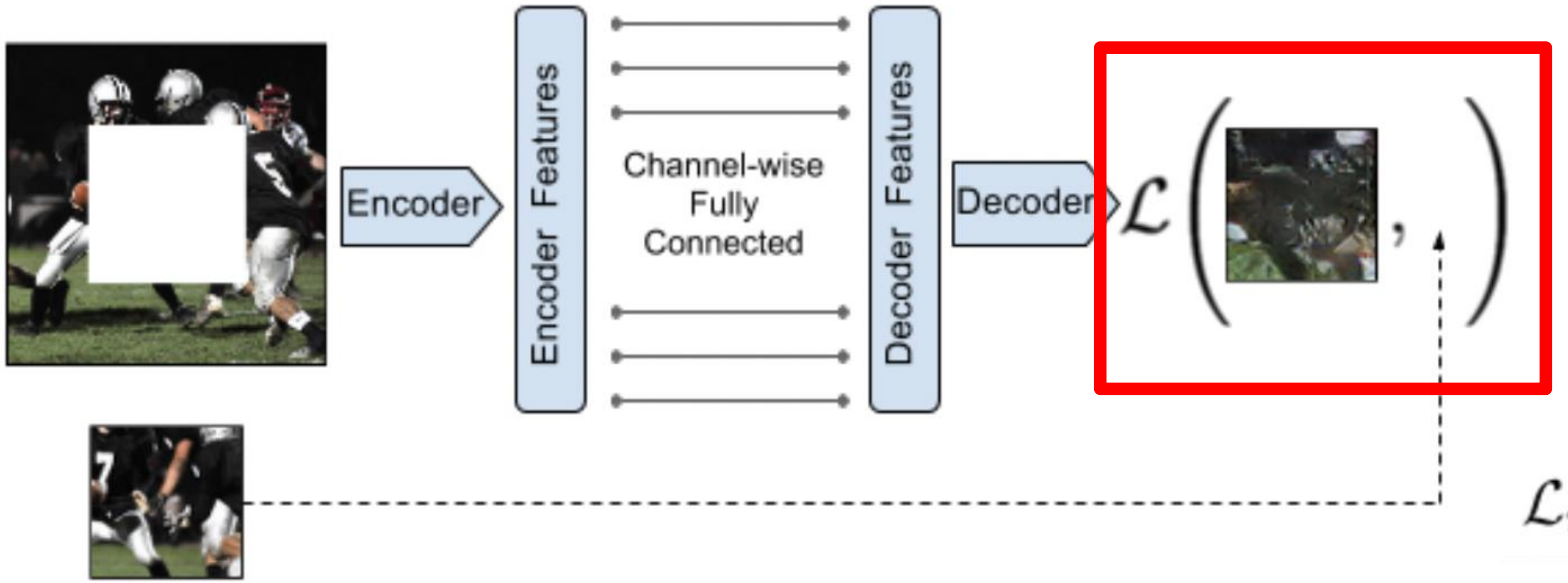
https://medium.com/knowledge-engineering-seminar/context-encoder-image-inpainting-using-gan-ccd6a1ea5fb7

Training: Loss Functions ($\mathcal{L} = \lambda_{adv}\mathcal{L}_{adv} + \lambda_{rec}\boxed{\mathcal{L}_{rec}}$)



https://medium.com/knowledge-engineering-seminar/context-encoder-image-inpainting-using-gan-ccd6a1ea5fb7

Training: Reconstruction Loss (i.e., Self-Supervised Learning Approach)



Learns to fit into the context by computing the L2 loss to compare the original patch content (P) to the predicted patch content created by the model when given the image with hole (CE(X')).

$$\mathcal{L}_{rec} = \|\mathbf{P} - CE(\mathbf{X}')\|_2^2$$

Pathak et al. Context encoders: Feature learning by inpainting; CVPR 2016

Training: Reconstruction Loss
(i.e., Self-Supervised Learning Approach)

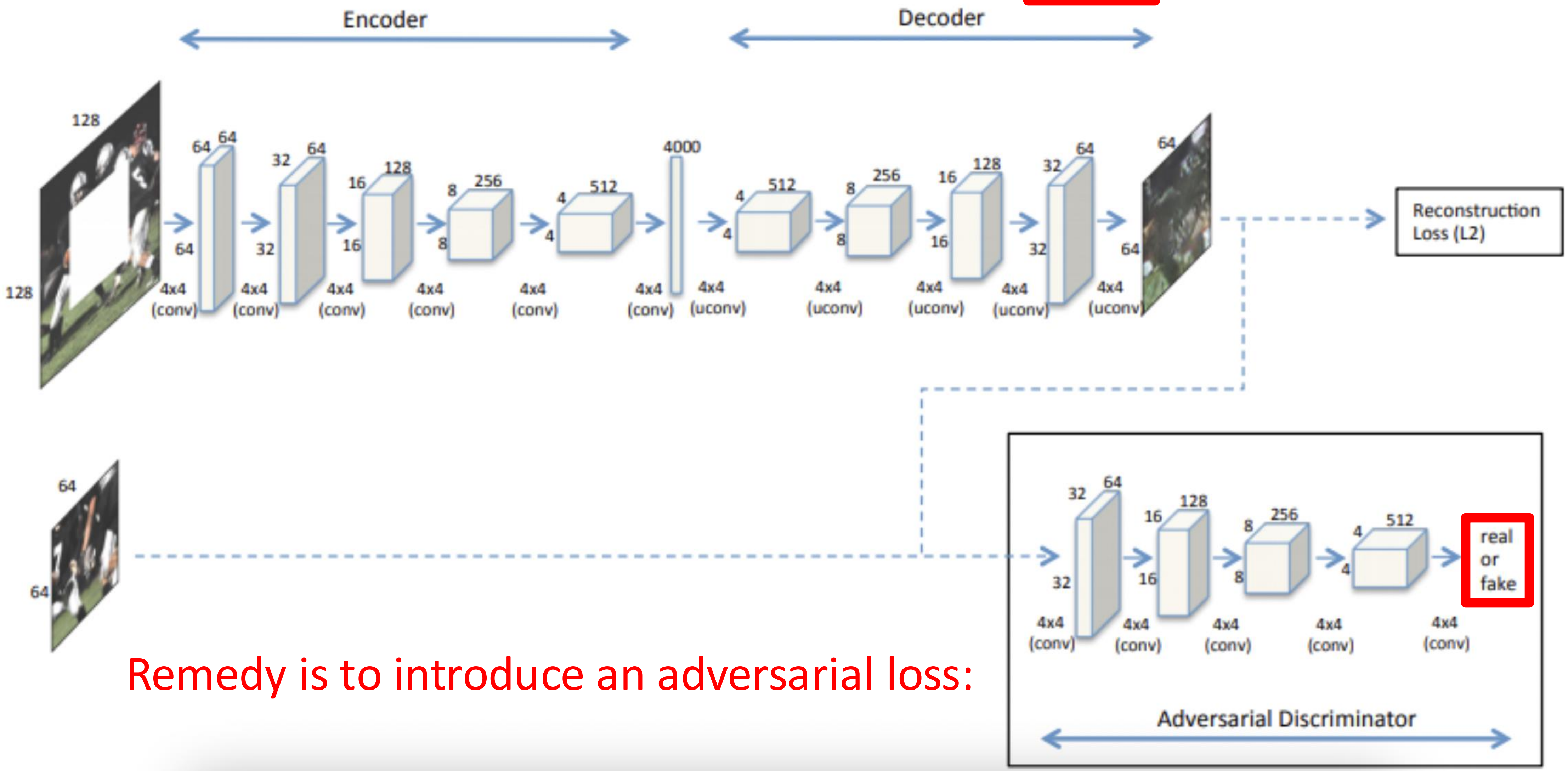
(a) Input context

(c) Context Encoder
($L2$ loss)

Why might training with this loss function alone lead to blurry results?
- It averages the multiple plausible inpaintings for a hole

Pathak et al. Context encoders: Feature learning by inpainting; CVPR 2016

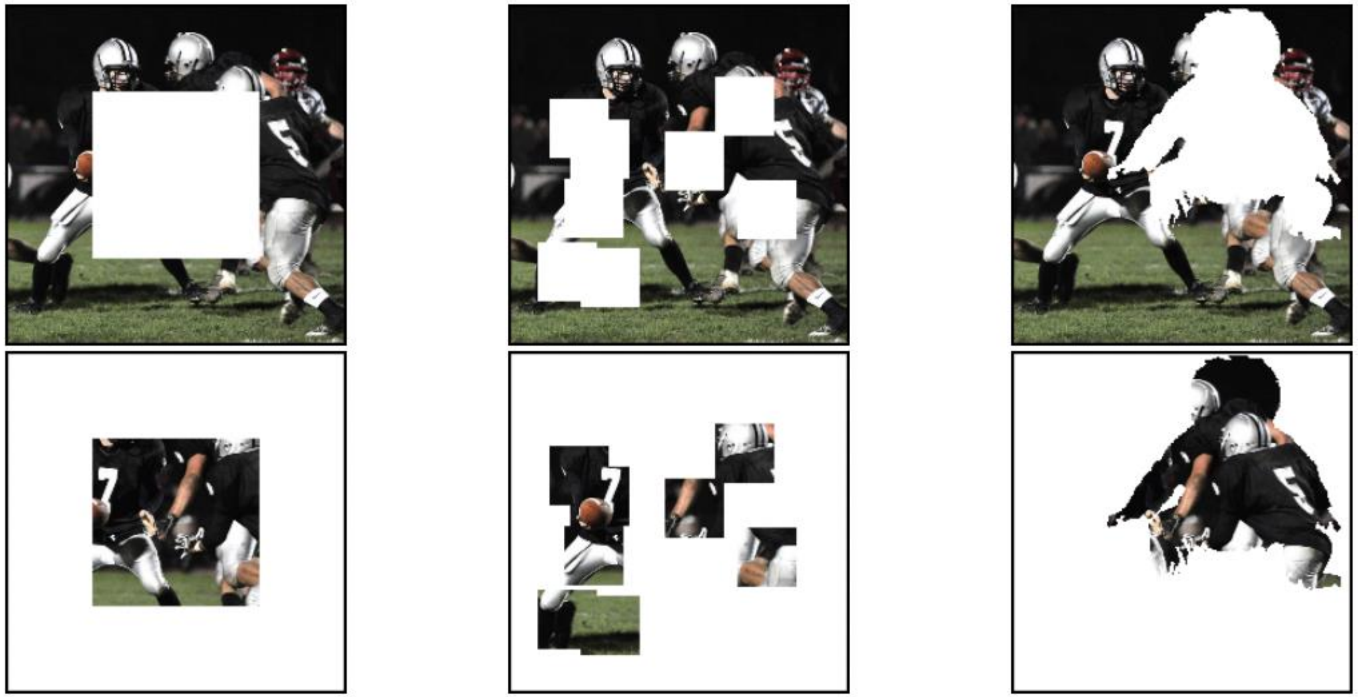
Training: Loss Functions ($\mathcal{L} = \lambda_{adv}\boxed{\mathcal{L}_{adv}} + \lambda_{rec}\mathcal{L}_{rec}$)



Remedy is to introduce an adversarial loss:

https://medium.com/knowledge-engineering-seminar/context-encoder-image-inpainting-using-gan-ccd6a1ea5fb7

Training: Datasets

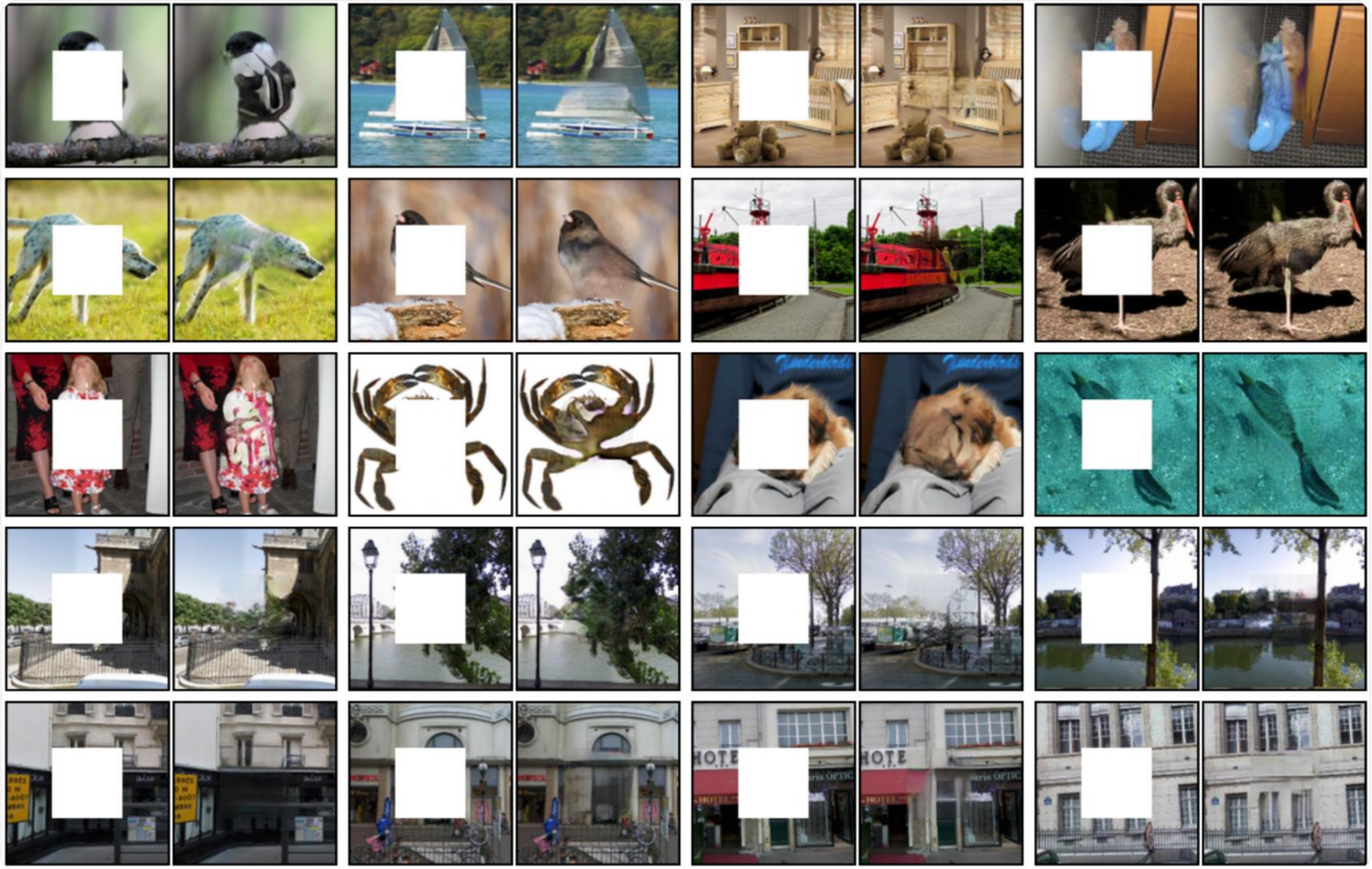


(a) Central region (b) Random block (c) Random region

Training completed on ImageNet (all 1.2M and a 100K subset) for three hole types

Pathak et al. Context encoders: Feature learning by inpainting; CVPR 2016

Results: https://www.cs.cmu.edu/~dpathak/context_encoder/



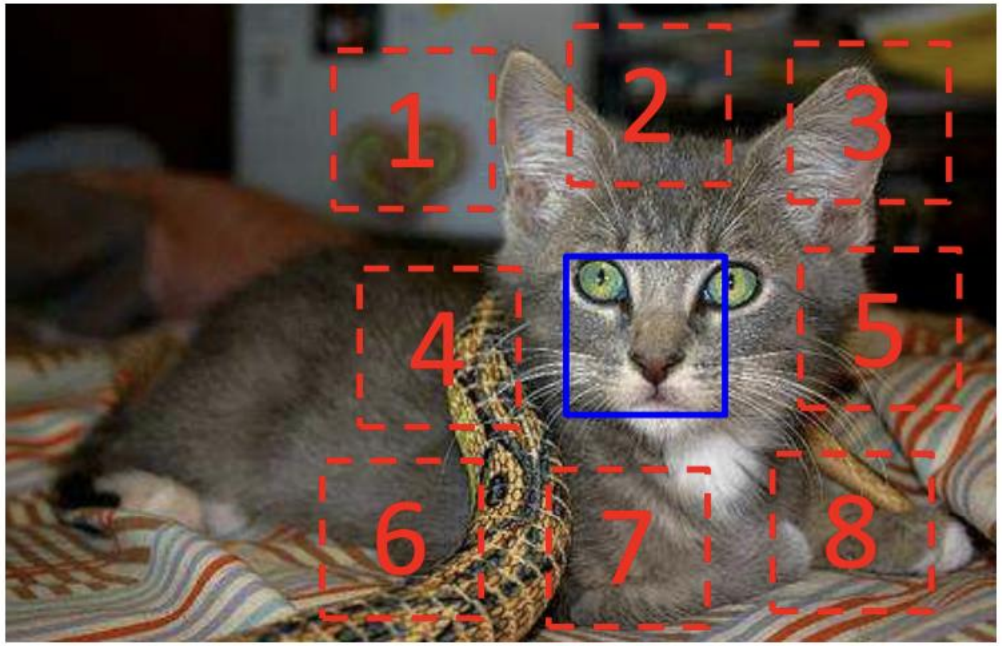
What type of features might be learned?

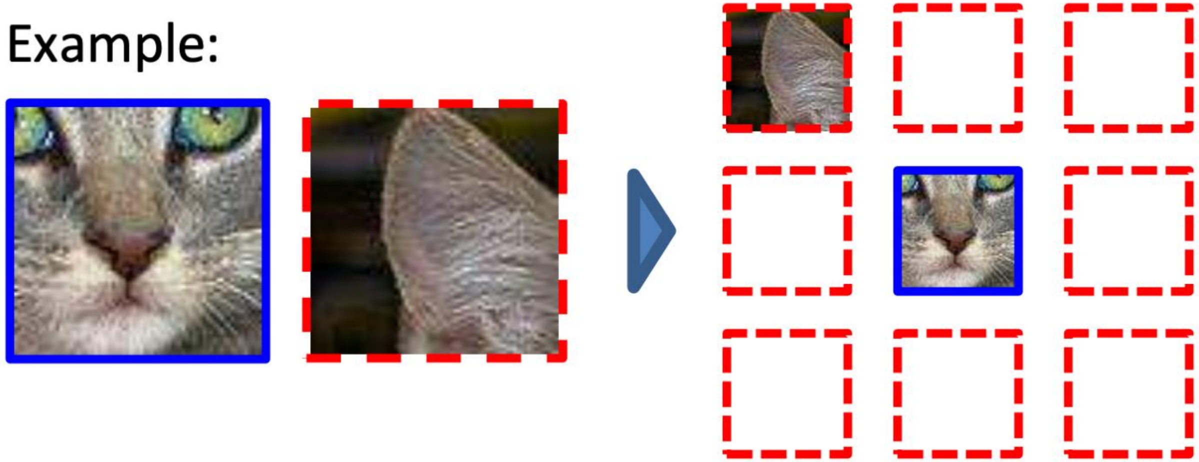
Pathak et al. Context encoders: Feature learning by inpainting; CVPR 2016

Types of Approaches

- Generative-based methods
- Generative adversarial networks
- Context-based methods

Spatial Context: Predict Image Index Per Patch

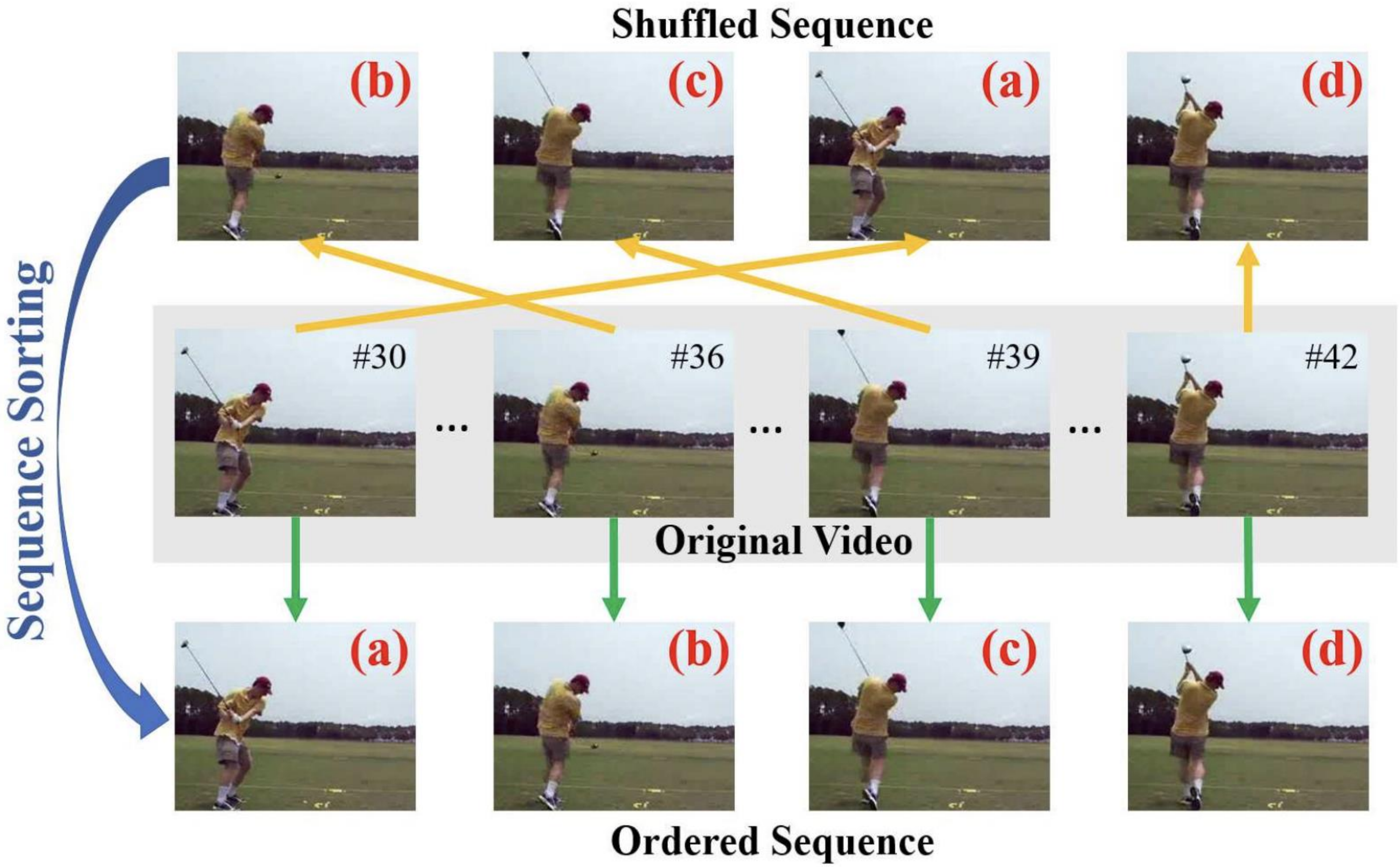




What type of features might be learned?

Doersch et al. Unsupervised Visual Representation Learning by Context Prediction. ICCV 2015.

Timing Context : Predict Order of Video Frames

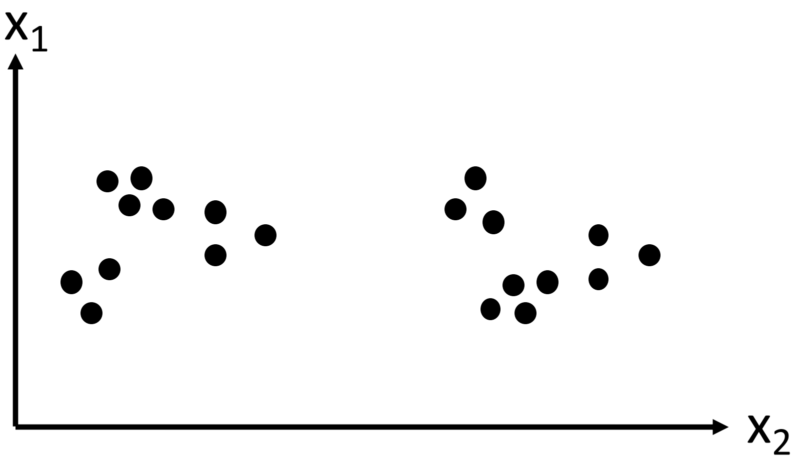


What type of features might be learned?

Lee et al. Unsupervised Representation Learning by Sorting Sequences. ICCV 2017.

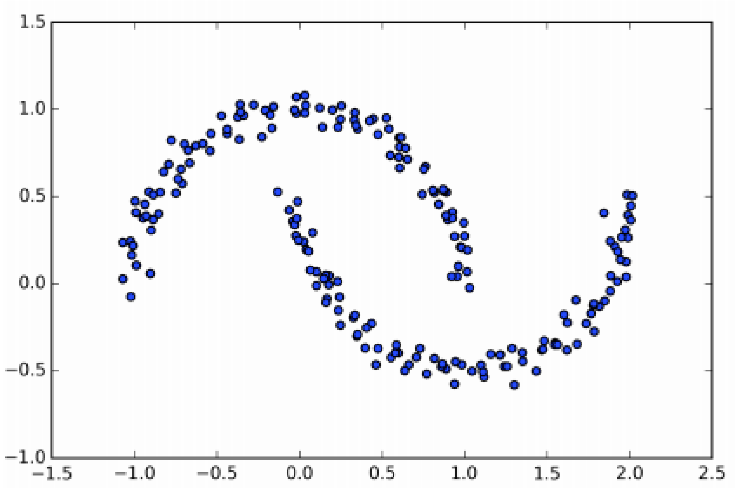
Similarity Context: Predict Clusters

A.



B.

C.

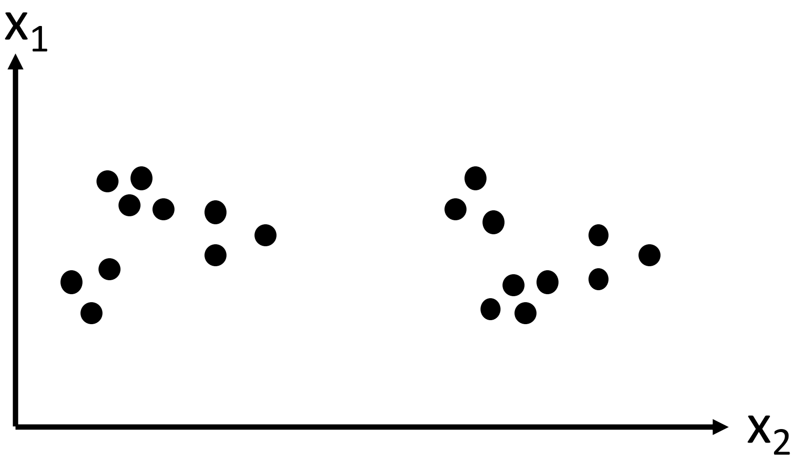


Models trained to identify cluster assignments OR to recognize whether images belong to the same cluster

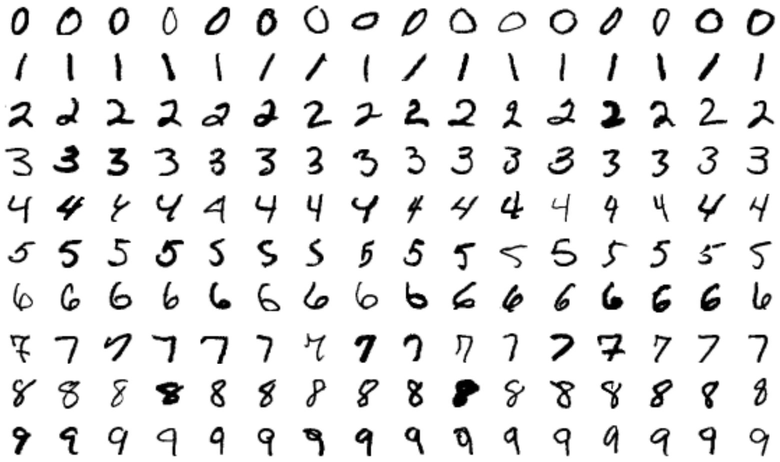
Raschka and Mirjalili; Python Machine Learning

Clustering

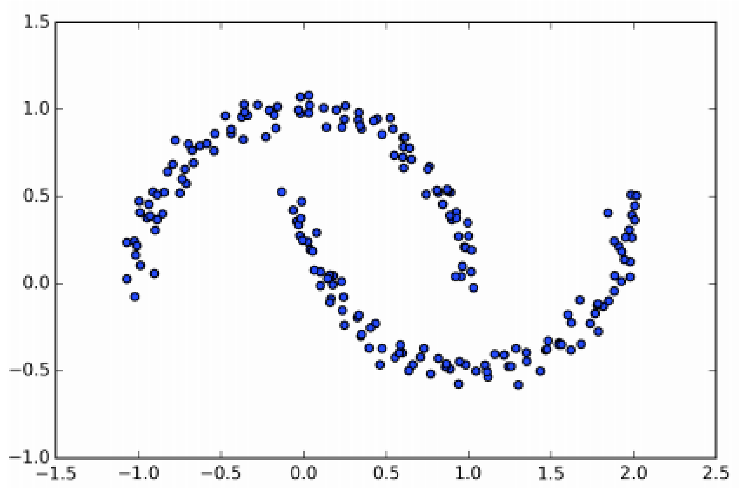
A.



B.



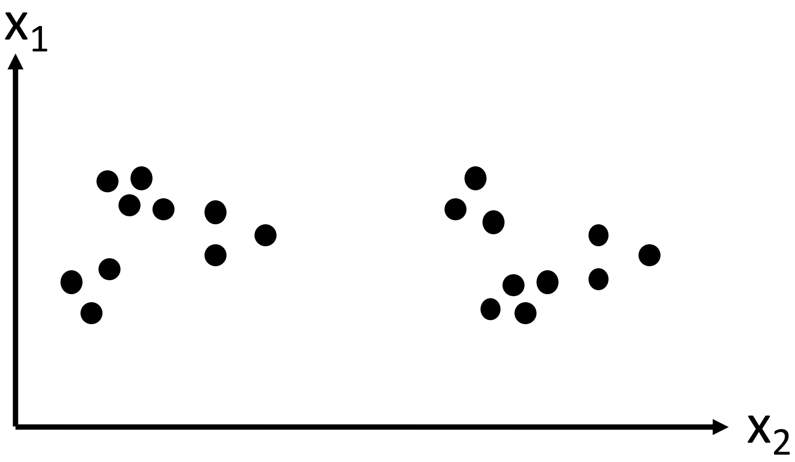
C.



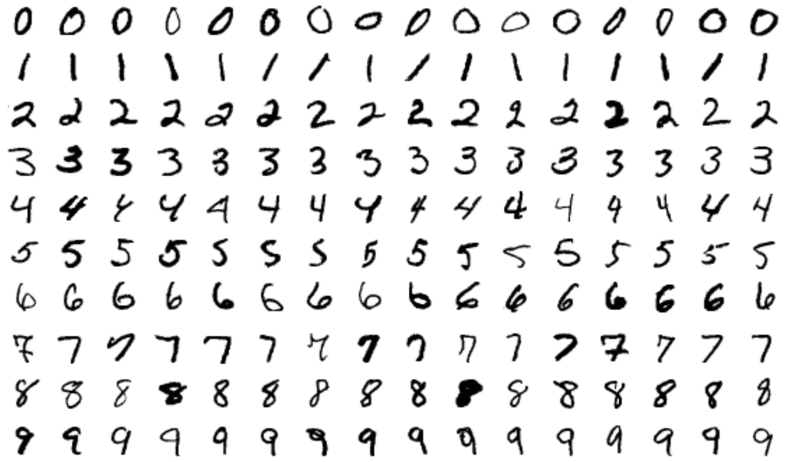
Create groupings so entities in a group will be similar to each other and different from the entities in other groups.

Clustering: Key Questions

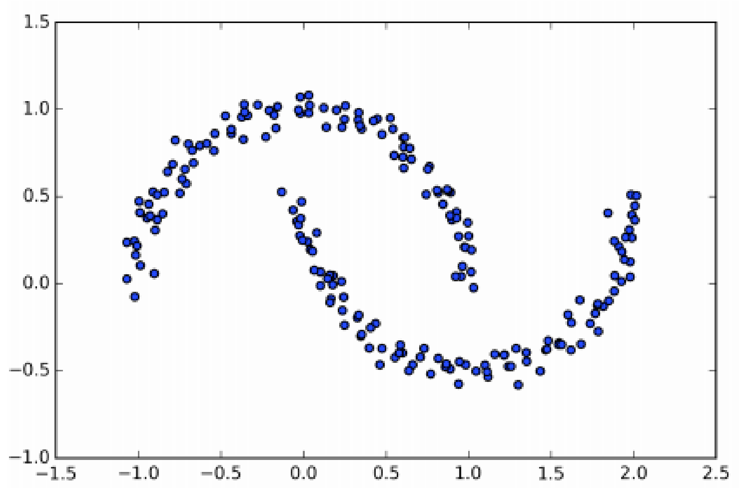
A.



B.



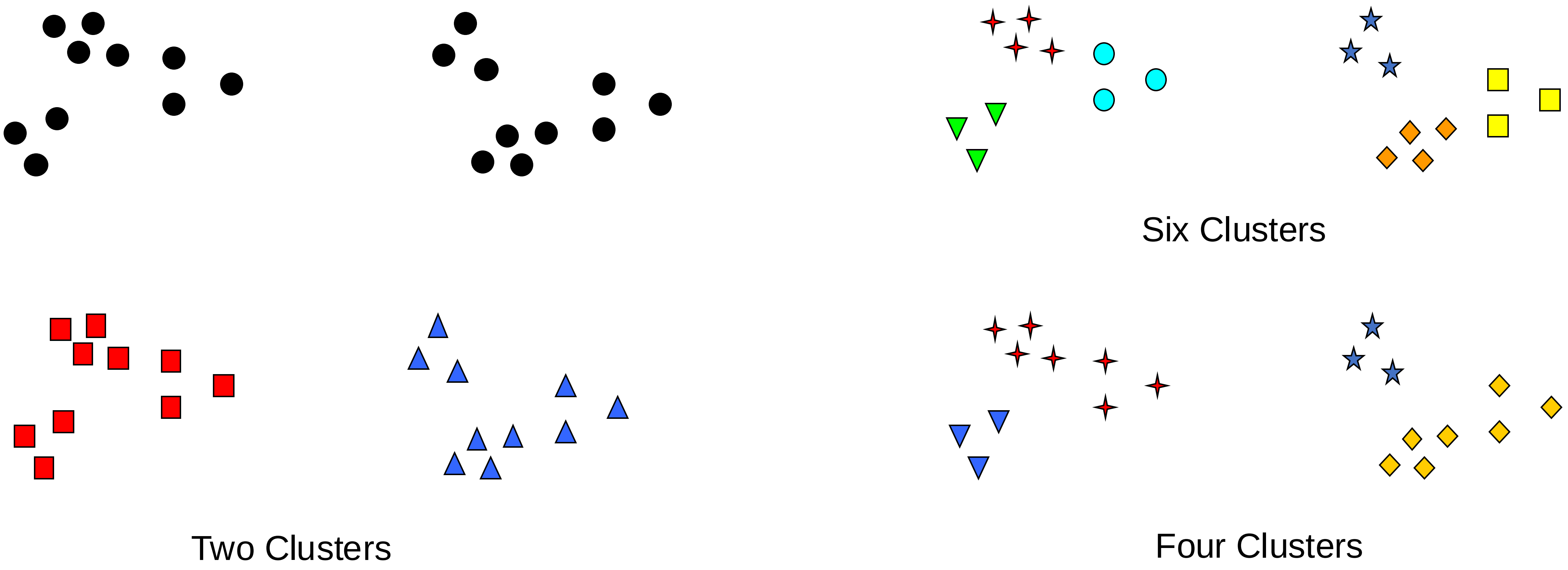
C.



- How many data clusters to create?
- What "algorithm" to use to partition the data?

Raschka and Mirjalili; Python Machine Learning

Clustering: How Many Clusters to Create?

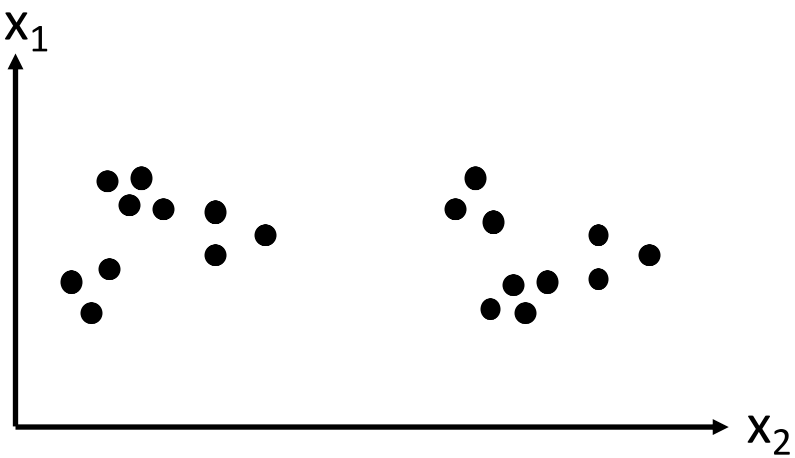


Number of clusters can be ambiguous

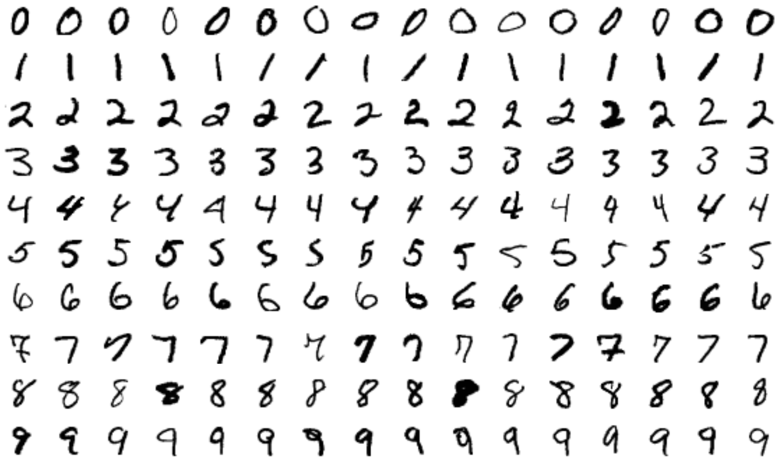
https://www-users.cs.umn.edu/~kumar001/dmbook/slides/chap7_basic_cluster_analysis.pdf

Clustering

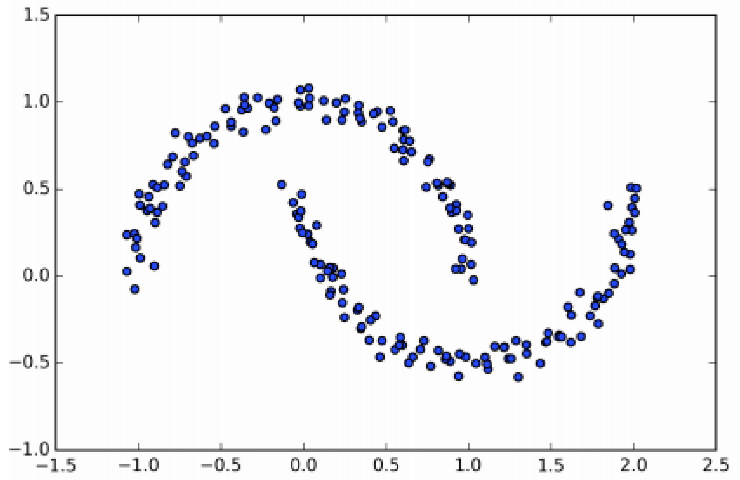
A.



B.



C.



Create groupings so entities in a group will be similar to each other and different from the entities in other groups.

What type of features might be learned?

Raschka and Mirjalili; Python Machine Learning

Summary: Many Types of Approaches

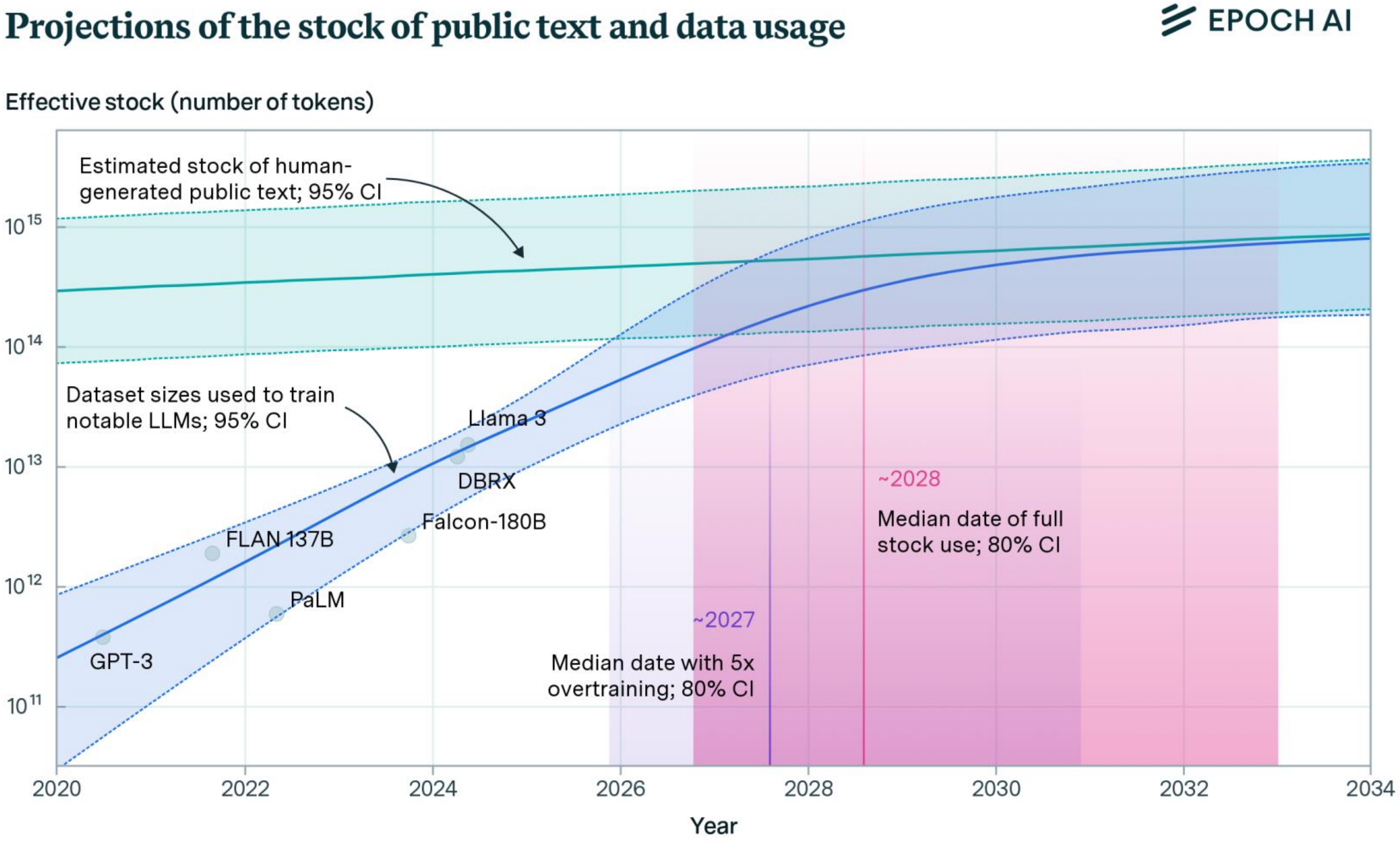
- Generative-based methods
- Generative adversarial networks
- Context-based methods

Today’s Topics

- Motivation
- Active learning
- Reinforcement learning
- Self-supervised learning
- And more…

Open Challenge 1: How to Scale When We Saturate Human-Generated Data?

- e.g., All public text data expected to be used some time between 2026 and 2032
- Increasingly popular idea: synthetic training data



https://epoch.ai/blog/will-we-run-out-of-data-limits-of-llm-scaling-based-on-human-generated-data

Open Challenge 2: How to Improve Data Quality?

- Increasingly popular idea: coreset selection (representative subset of dataset)

- Feb 2025 media-catching example: s1
  - outperformed OpenAI's o1
  - Qwen2.5 base model fine-tuned on 1,000 selected questions and corresponding Gemini-generated answers, which cost ~$20 to $50 in cloud compute credits.
  - *It also benefited from appending "Wait", so the model continues with self-review*

Today’s Topics

- Motivation
- Active learning
- Coreset selection
- Self-supervised learning
- Synthetic training data


The End