Carnegie Mellon University
HeinzCollege

Interpreting What Deep Nets are Learning, Other Deep Learning Topics, Wrap-up

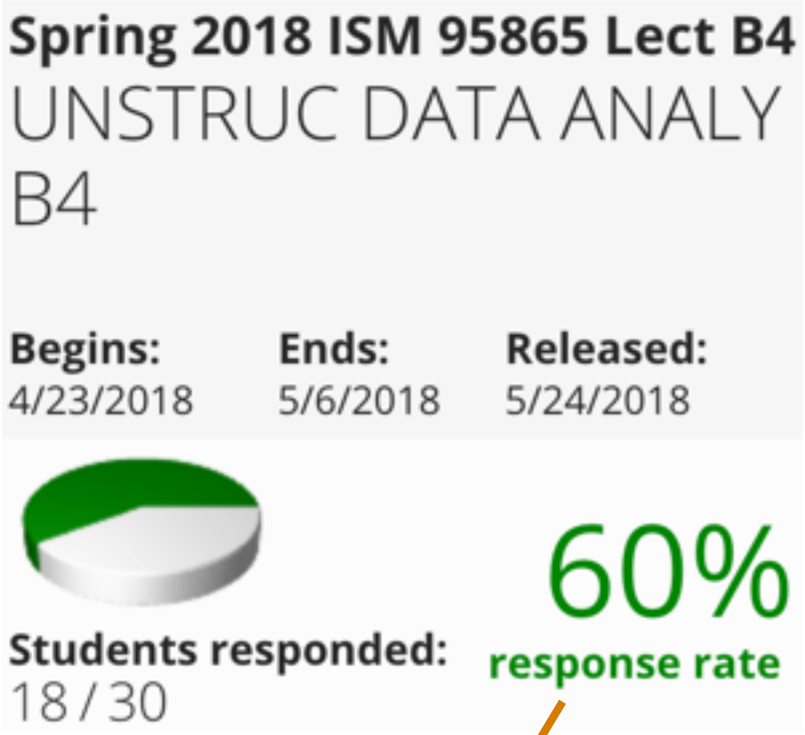
George Chen

Final Exam

- Next Tuesday 5/8, 1pm-4pm
    - Bring a laptop to work on (Jupyter notebook)
        - You are responsible for making sure your laptop is working properly, has enough battery life, etc
        - No AWS
    - Open book: you may use course materials, API documentation, stackoverflow, etc (be sure to cite external resources like stackoverflow if you use it)
    - No collaboration (so obviously do not post to Piazza)

Faculty Course Evaluations

- Please answer these to provide valuable feedback/vent your frustration



Spring 2018 ISM 95865 Lect B4

UNSTRUC DATA ANALY B4

Begins: 4/23/2018 **Ends:** 5/6/2018 **Released:** 5/24/2018

Students responded: 18 / 30

60% **response rate**

🙁

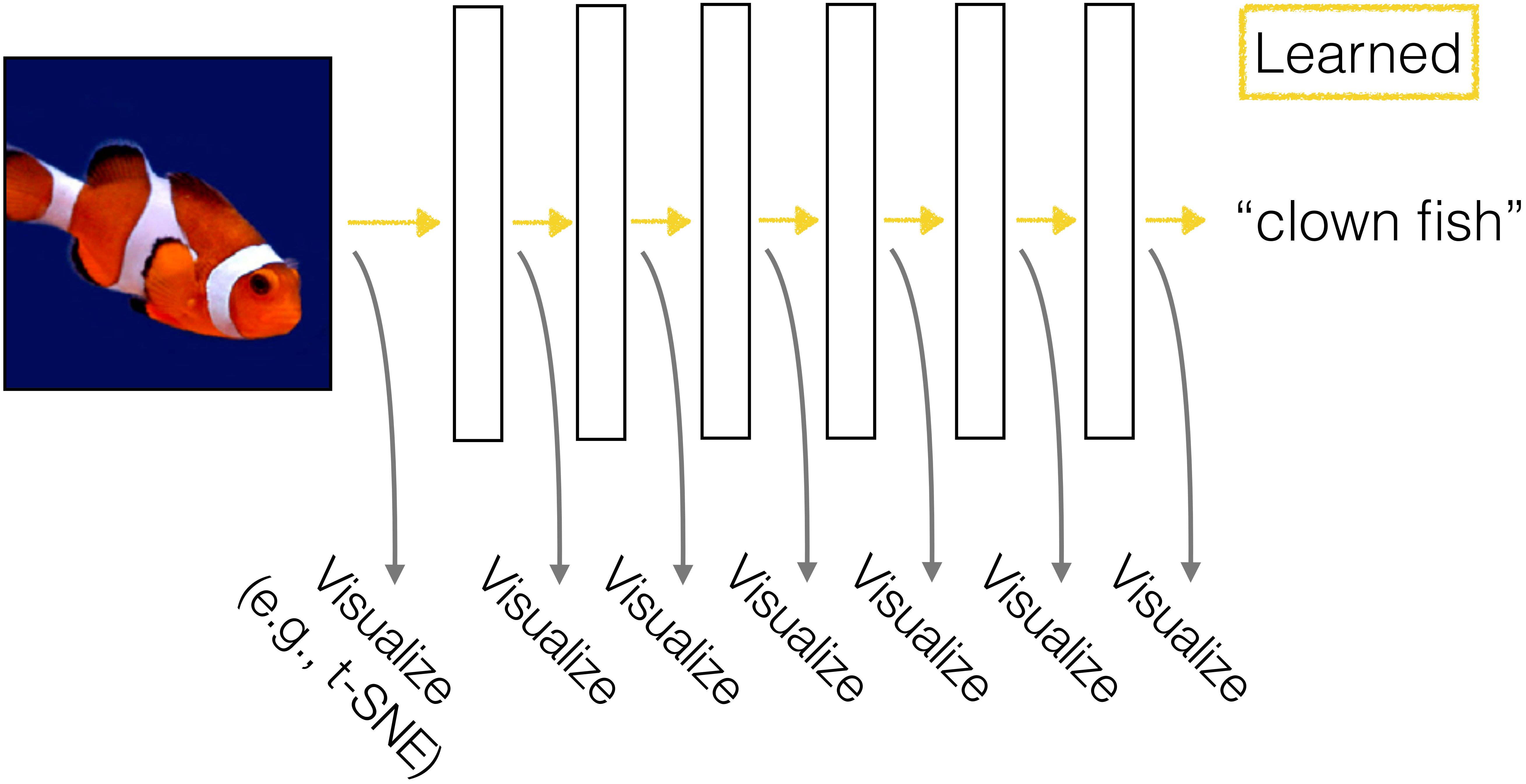
Today

- Interpreting what a deep net is learning

- High-level overview of a bunch of deep learning topics we didn’t get to

- Course wrap-up

What is a Deep Net Learning?



Learned

“clown fish”

Visualize (e.g., t-SNE)

Visualize

Visualize

Visualize

Visualize

Visualize

Visualize

1 strategy: just put in test images and visualize intermediate outputs

Another Strategy

How much does an output neuron depend on a specific input?

$$f(x, y) = x^2 + xy$$

$$\frac{\partial f(x, y)}{\partial x} = 2x + y$$

$$\frac{\partial f(x, y)}{\partial y} = x$$

For specific inputs *x* and *y:* look at how large these derivatives are!

Another Strategy

How much does an output neuron depend on a specific input?

$$f(x, y) = x^2 + xy$$

$$\frac{\partial f(x, y)}{\partial x} = 2x + y \quad = 2\ (0) + 0.5 = 0.5$$

e.g., *x* = 0, *y* = 0.5

$$\frac{\partial f(x, y)}{\partial y} = x \quad = 0$$

Conclude: in this case, *x* has larger effect on output than *y*

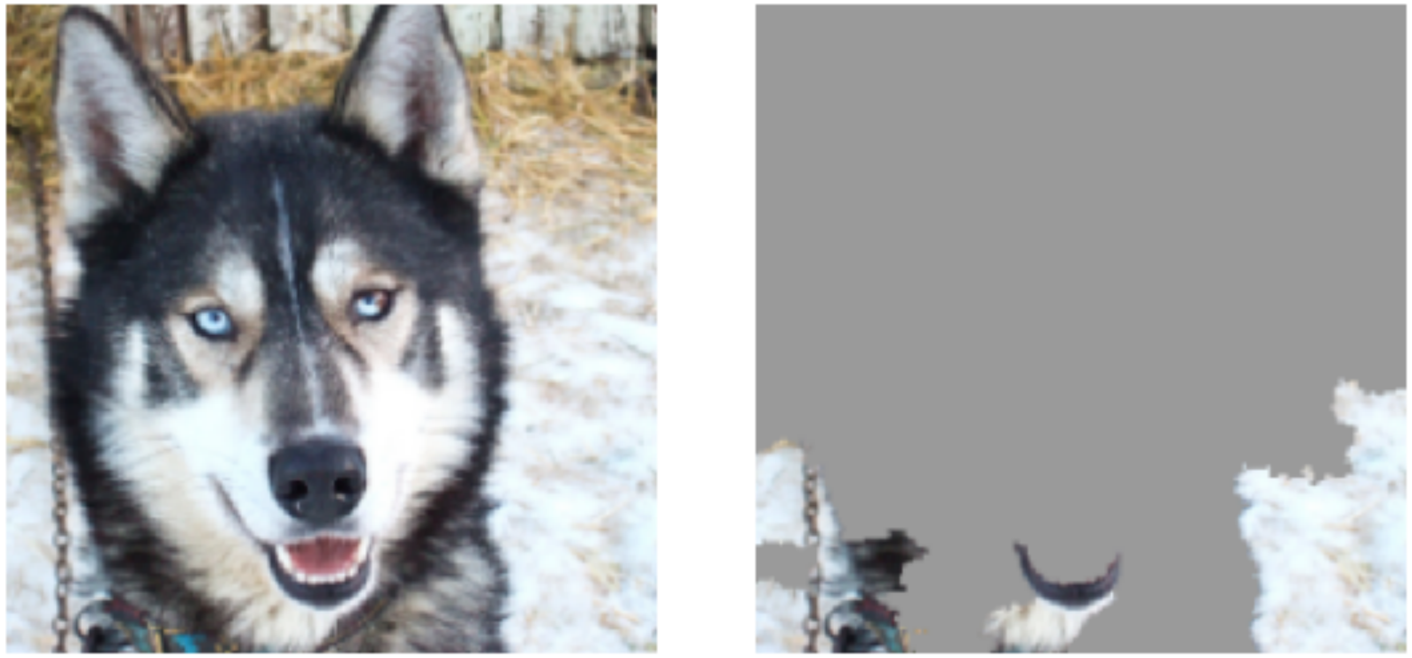
For specific inputs *x* and *y:* look at how large these derivatives are!

For any two neurons, we can look at how much one affects another

Interpreting Deep Nets

Demo

Example: Wolves vs Huskies



(a) Husky classified as wolf

(b) Explanation

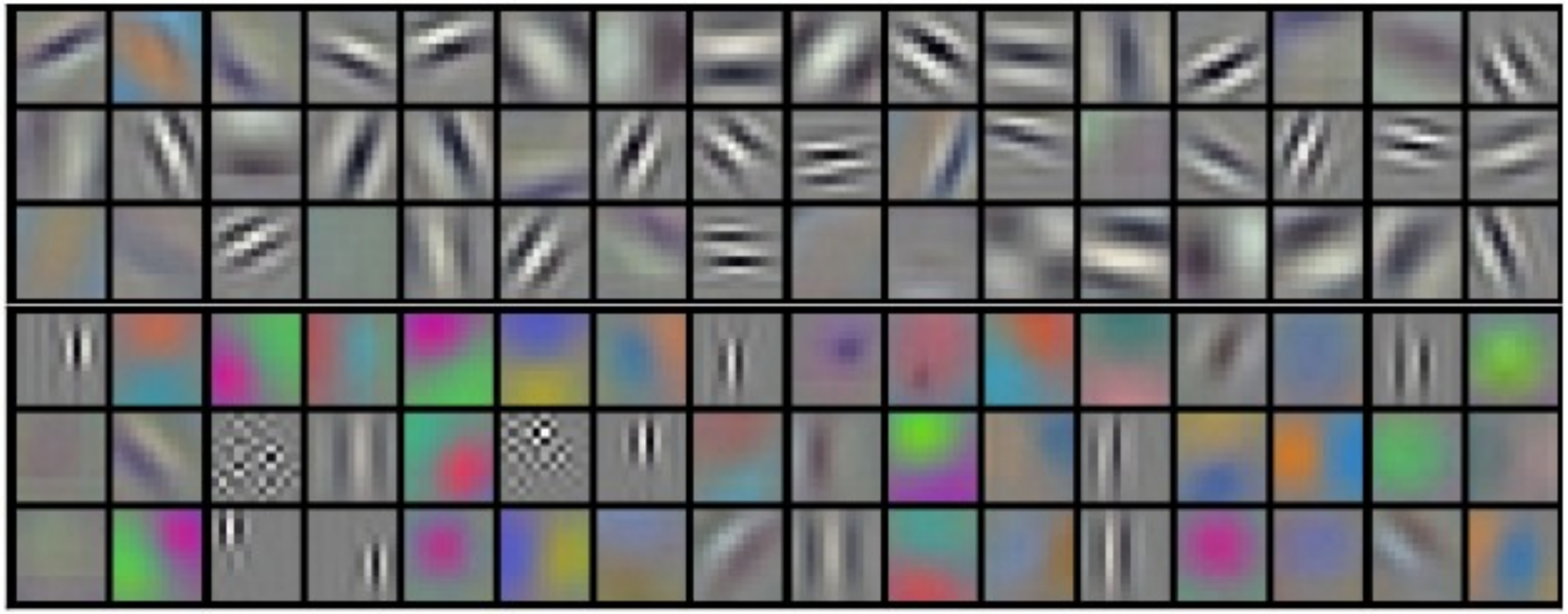
Turns out the deep net learned that wolves are wolves because of snow…

➔ visualization is crucial!

Source: Ribeiro et al. “Why should I trust you? Explaining the predictions of any classifier.” KDD 2016.

Example: Learned Filters

It’s also possible to visualize filters of convolutional layers



Source: http://cs231n.github.io/convolutional-networks/

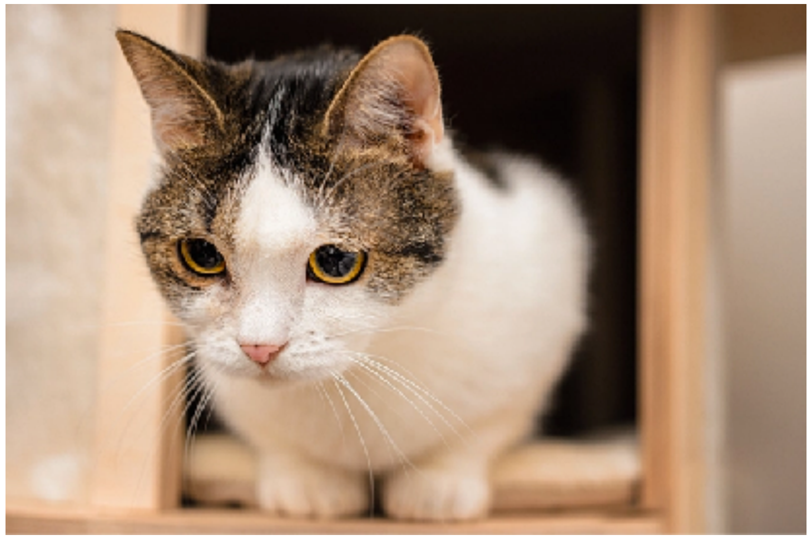
There’s a lot more to deep learning that we didn’t cover

Dealing with Small Datasets

Data augmentation: generate perturbed versions of your training data to get larger training dataset

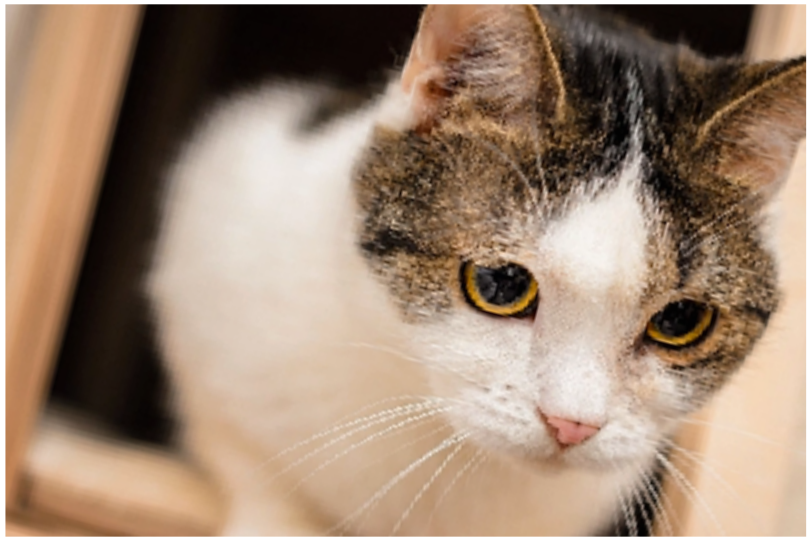
Training image

Training label: cat



Mirrored

Still a cat!



Rotated & translated

Still a cat!

We just turned 1 training example in 3 training examples

Allowable perturbations depend on data (e.g., for handwritten digits, rotating by 180 degrees would be bad: confuse 6's and 9's)

Dealing with Small Datasets

Fine tuning: if there's an existing pre-trained neural net, you could modify it for your problem that has a small dataset

Example: classify between Tesla's and Toyota's

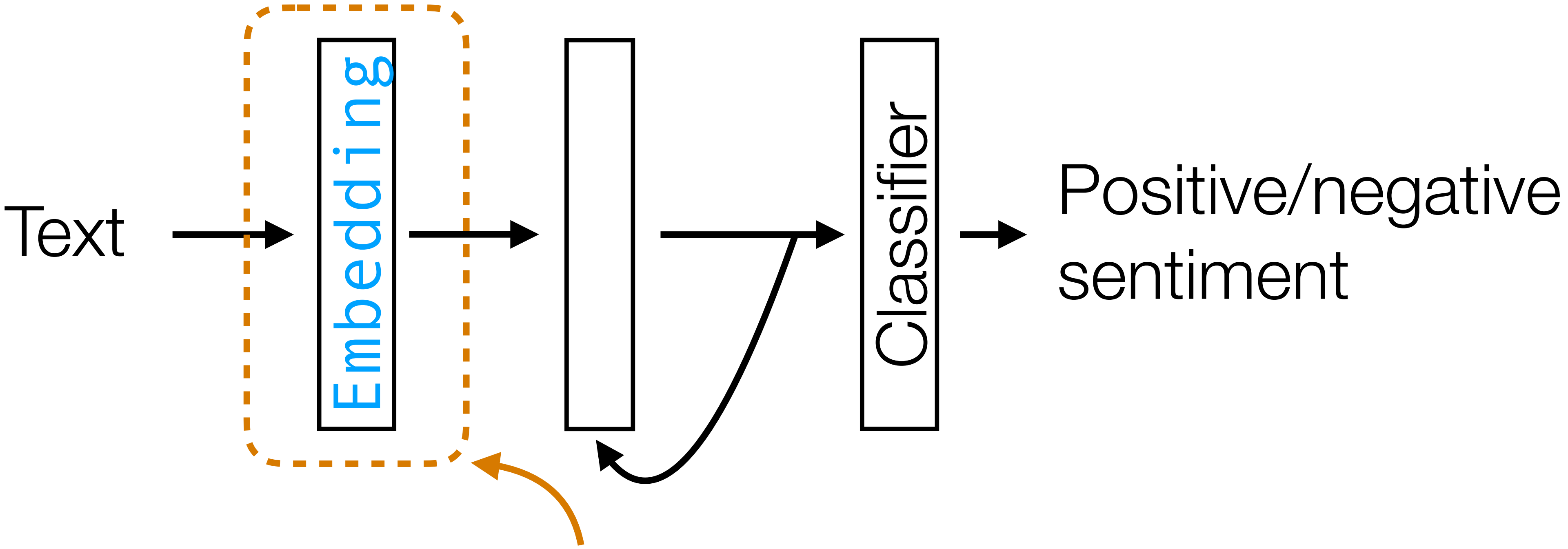
You collect photos from the internet of both, but your dataset size is small, on the order of 1000 images

Strategy: take existing pre-trained CNN for ImageNet classification and change final layer to do classification between Tesla's and Toyota's rather than classifying into 1000 objects

Dealing with Small Datasets

Fine tuning: if there's an existing pre-trained neural net, you could modify it for your problem that has a small dataset

Example: sentiment analysis RNN demo



We fixed the weights here to come from GloVe and disabled training for this layer!

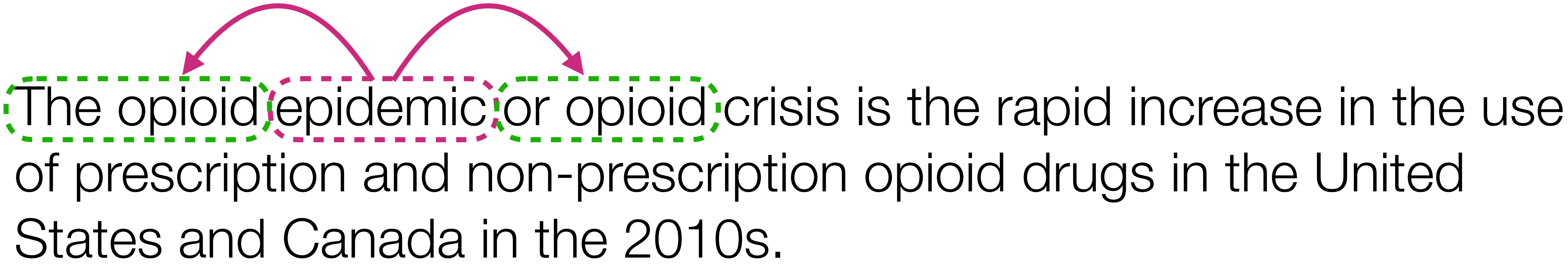
GloVe vectors pre-trained on massive dataset (Wikipedia + Gigaword)

IMDb review dataset is small in comparison

Self-Supervised Learning

Even without labels, we can set up a prediction task!

Example: word embeddings like word2vec, GloVe



The opioid epidemic or opioid crisis is the rapid increase in the use of prescription and non-prescription opioid drugs in the United States and Canada in the 2010s.

Predict context of each word!

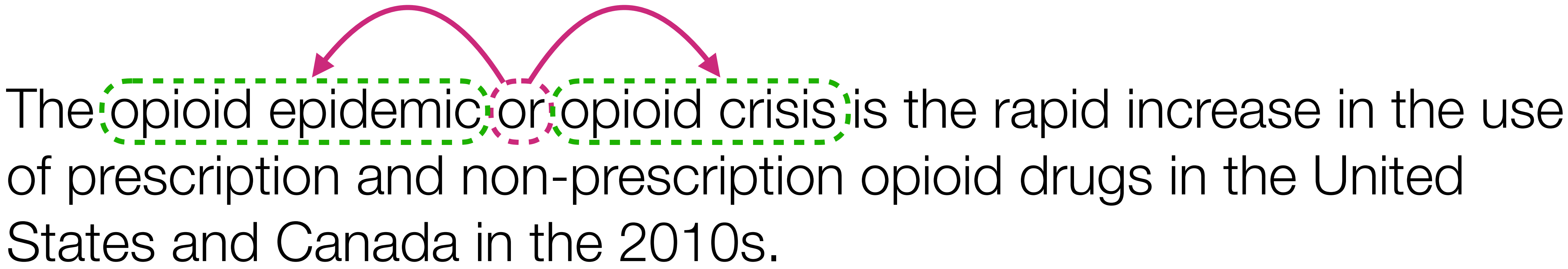
Training data point: epidemic

“Training label”: the, opioid, or, opioid

Self-Supervised Learning

Even without labels, we can set up a prediction task!

Example: word embeddings like word2vec, GloVe



The opioid epidemic or opioid crisis is the rapid increase in the use of prescription and non-prescription opioid drugs in the United States and Canada in the 2010s.

Predict context of each word!

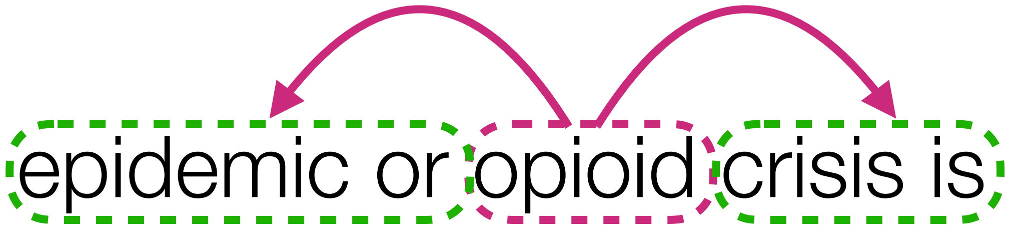
Training data point: or

“Training label”: opioid, epidemic, opioid, crisis

Self-Supervised Learning

Even without labels, we can set up a prediction task!

Example: word embeddings like word2vec, GloVe



The opioid epidemic or opioid crisis is the rapid increase in the use of prescription and non-prescription opioid drugs in the United States and Canada in the 2010s.

Predict context of each word!

There are "positive" examples of what context words are for "opioid"

Training data point: opioid

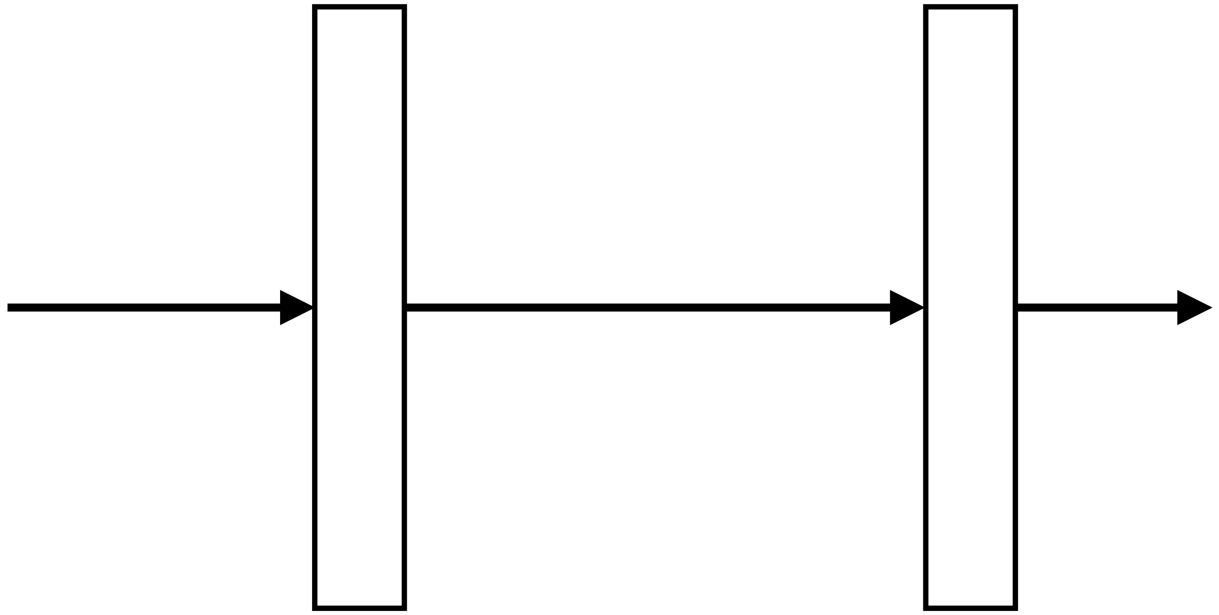
"Training label": epidemic, or, crisis, is

Also provide "negative" examples of words that are *not* likely to be context words (e.g., randomly sample words elsewhere in document)

Self-Supervised Learning

Even without labels, we can set up a prediction task!

Example: word embeddings like word2vec, GloVe



Input word (categorical "one hot" encoding)

Vector saying the probabilities of different words being context words

Dense layer
# neurons equal to embedding dim.

Dense layer,
softmax activation

This actually relates to PMI!

Weight matrix: (# words in vocab) by (embedding dim)

Dictionary word *i* has "word embedding" given by row *i* of weight matrix

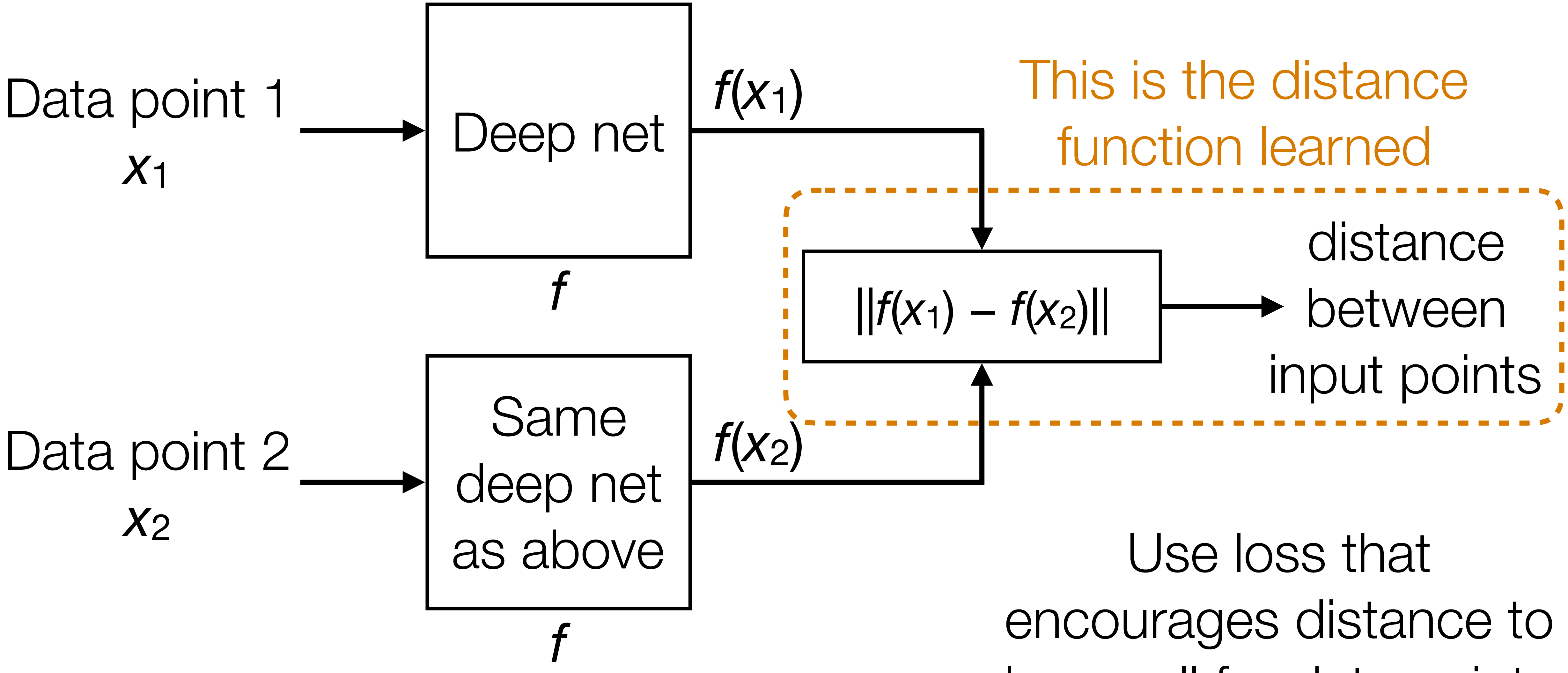
Self-Supervised Learning

Even without labels, we can set up a prediction task!

- Key idea: predict part of the training data from other parts of the training data
- No actual training labels required — we are defining what the training labels are just using the unlabeled training data
- This is an *unsupervised* method that sets up a *supervised prediction* task

Learning Distances with Siamese Nets

Using labeled data, we can learn a distance function



This is the distance function learned

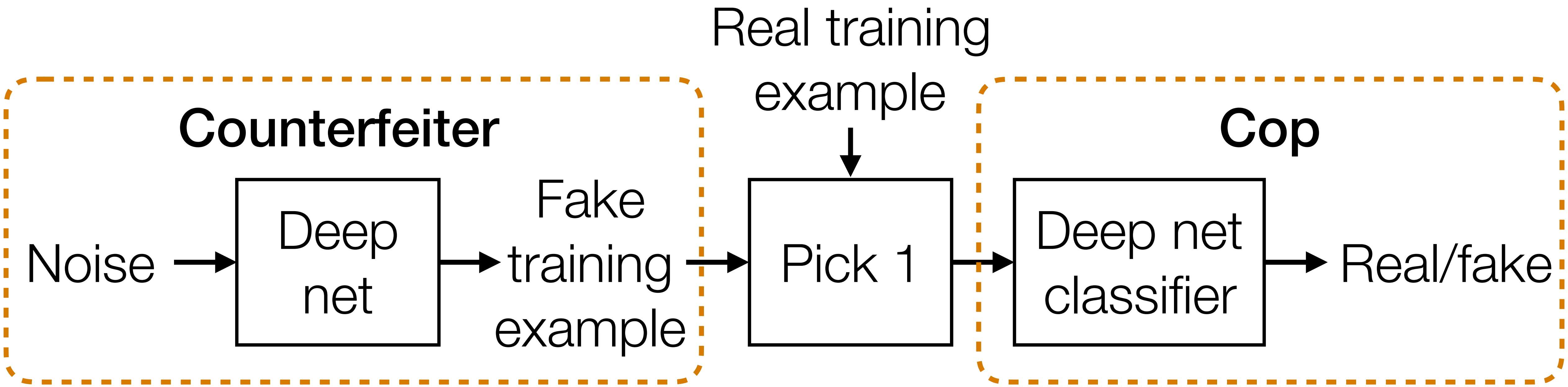
Use loss that encourages distance to be small for data points with same label and large otherwise

Note: we are learning the function *f*

Generate Fake Data that Look Real

Unsupervised approach: generate data that look like training data

Example: Generative Adversarial Network (GAN)



Counterfeiter tries to get better at tricking the cop

Cop tries to get better at telling which examples are real vs fake

Terminology: counterfeiter is the **generator**, cop is the **discriminator**

Other approaches: variational autoencoders, pixelRNNs/pixelCNNs

Generate Fake Data that Look Real

Fake celebrities generated by NVIDIA using GANs
(Karras et al Oct 27, 2017)

Google DeepMind’s WaveNet makes fake audio that sounds like whoever you want using pixelRNNs (Oord et al 2016)

Generate Fake Data that Look Real

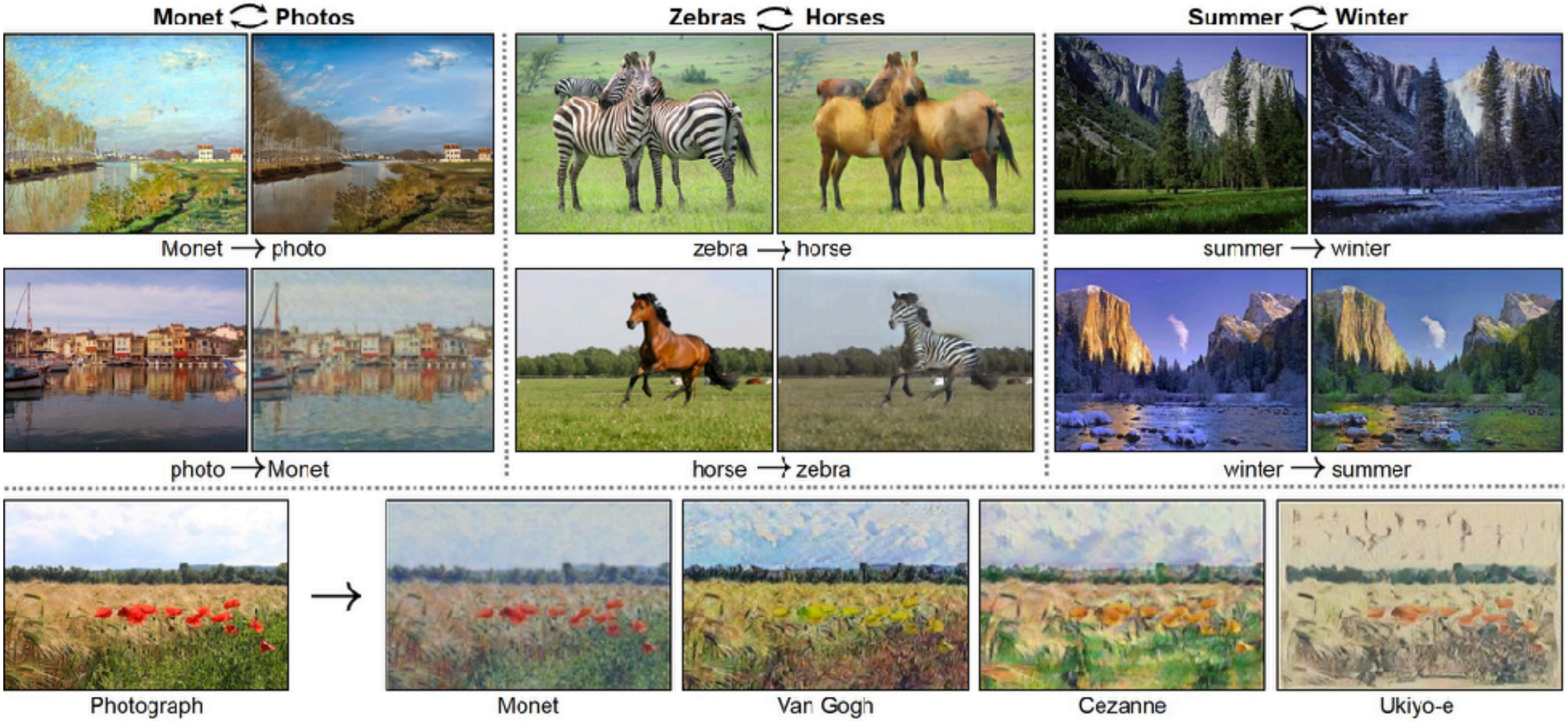
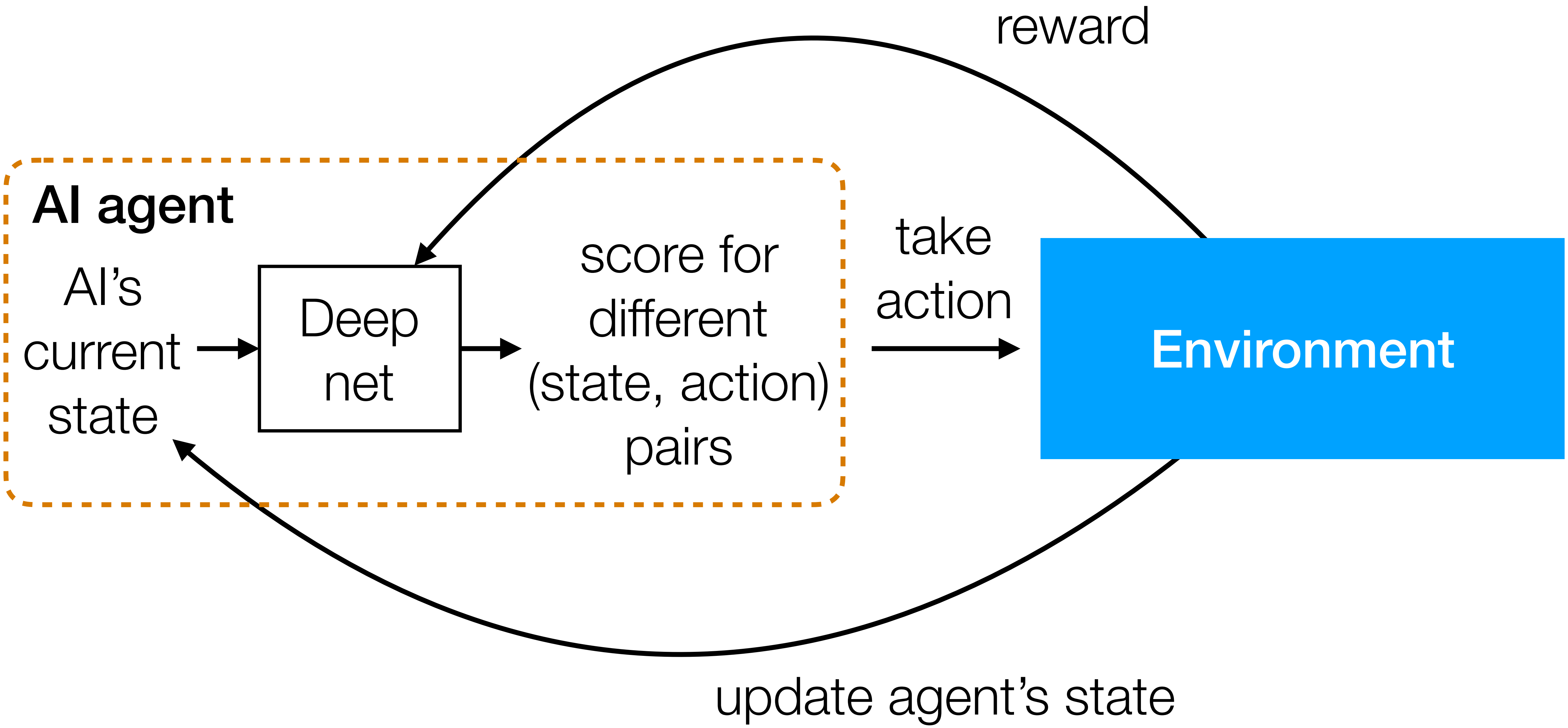


Image-to-image translation results from UC Berkeley using GANs (Isola et al 2017, Zhu et al 2017)

Deep Reinforcement Learning

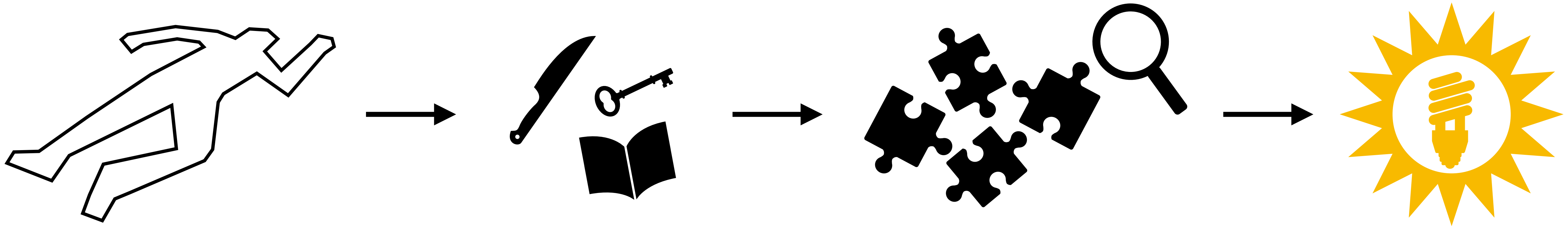
The machinery behind AlphaGo and similar systems



The Future of Deep Learning

- Deep learning currently is still limited in what it can do — the layers do simple operations and have to be differentiable
    - How do we make deep nets that generalize better?
- Still lots of engineering and expert knowledge used to design some of the best systems (e.g., AlphaGo)
    - How do we get away with using less expert knowledge?
- How do we do lifelong learning?

Unstructured Data Analysis



Question

The dead body

This is provided by a practitioner

Data

The evidence

Some times you have to collect more evidence!

Finding Structure

Puzzle solving, careful analysis

Exploratory data analysis

Insights

When? Where? Why? How? Perpetrator catchable?

Answer original question

There isn’t always a follow-up prediction problem to solve

95-865 Some Parting Thoughts

- Remember to **visualize different steps of your data analysis pipeline**
  - Helpful for both debugging and interpreting final output!
- Very often there are *tons* of models/design choices to try
  - Come up with **quantitative metrics** that make sense for your problem, and use these metrics to **evaluate models with a prediction task on held-out data**
  - But don't blindly rely on metrics without **interpreting results in the context of your original problem**!
- Often times you won't have labels!
  - Manually obtain labels (either you do it or crowdsource)
  - Set up self-supervised learning task