Deep Learning

Philosophy, Working Principles and Algorithms

Presented by

Fayyaz Minhas

Department of Computer Science
University of Warwick

@fayyazhere

Philosophy

Train PhD students to be thinkers not just specialists

Many doctoral curricula aim to produce narrowly focused researchers rather than critical thinkers. That can and must change, says **Gundula Bosch**.

Under pressure to turn out productive lab members quickly, many PhD programmes in the biomedical sciences have shortened their courses, squeezing out opportunities for putting research into its wider context. Consequently, most PhD curricula are unlikely to nurture the big thinkers and creative problem-solvers that society needs.

That means students are taught every detail of a microbe's life cycle but little about the life scientific. They need to be taught to recognize how errors can occur. Trainees should evaluate case studies derived from flawed real research, or use interdisciplinary detective games to find logical fallacies in the literature. Above all, students must be shown the scientific process as it is — with its limitations and potential pitfalls as well as its fun side, such as serendipitous discoveries and hilarious blunders.

This is exactly the gap that I am trying to fill at Johns Hopkins University in Baltimore, Maryland, where a new graduate science programme is entering its second year. Microbiologist Arturo Casadevall and I began pushing for reform in early 2015, citing the need to put the philosophy back into the doctorate of philosophy: that is, the 'Ph' back into the PhD. We call our programme R3, which means that our students learn to apply rigour to their design and conduct of experiments; view their work through the lens of social responsibility; and to think critically, communicate better, and thus improve reproducibility. Although we are aware of many innovative individual courses developed along these lines, we are striving for more-comprehensive reform.

PUT THE **PHILOSOPHY** BACK INTO THE **DOCTORATE** OF PHILOSOPHY.

Our offerings are different from others at the graduate level. We have critical-thinking assignments in which students analyse errors in reasoning in a *New York Times* opinion piece about 'big sugar', and the ethical implications of the arguments made in a *New Yorker* piece by surgeon Atul Gawande entitled 'The Mistrust of Science'. Our courses on rigorous research, scientific integrity, logic, and mathematical and programming skills are integrated into students' laboratory and fieldwork. Those studying the influenza virus, for example, work with real-life patient data sets and wrestle with the challenges of applied statistics.

A new curriculum starts by winning allies. Both students and faculty members must see value in moving off the standard track. We used informal interviews and focus groups to identify areas in which students and faculty members saw gaps in their training. Recurring themes included the inability to apply theoretical knowledge in statistical tests in the laboratory, frequent mistakes in choosing an appropriate set of experimental controls, and significant difficulty in explaining work to non-experts.

Introducing our programme to colleagues in the Johns Hopkins life-sciences departments was even more sensitive. I was startled by the oft-expressed opinion that scientific productivity depended more on rote knowledge than on competence in critical thinking. Several principal investigators were uneasy about students committing more time to less conventional forms of education. The best way to gain their support was coffee: we repeatedly met lab heads to understand their concerns.

With the pilot so new, we could not provide data on students' performance, but we could address faculty members' scepticism. Some colleagues were apprehensive that students would take fewer courses in specialized content to make room for interdisciplinary courses on ethics, epistemology and quantitative skills. In particular, they worried that the R3 programme could lengthen the time required for students to complete their degree, leave them insufficiently knowledgeable in their subject areas and make them less productive in the lab.

We made the case that better critical thinking and fewer mandatory discipline-specific classes might actually position students to be more productive. We convinced several professors to try the new system and participate in structured evaluations on whether R3 courses contributed to students' performance.

So far, we have built 5 new courses from scratch and have enrolled 85 students from nearly a dozen departments and divisions. The courses cover the anatomy of errors and misconduct in scientific practice and teach students how to dissect the scientific literature. An interdisciplinary discussion series encourages broad and critical thinking about science. Our students learn to consider societal consequences of research advances, such as the ability to genetically alter sperm and eggs.

Discussions about the bigger-picture problems of the scientific enterprise get students to reflect on the limits of science, and where science's ability to do something competes with what scientists should do from a moral point of view. In addition, we have seminars and workshops on professional skills, particularly leadership skills through effective communication, teaching and mentoring.

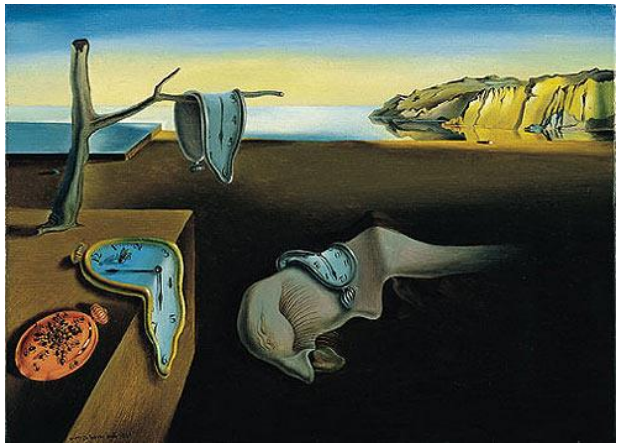
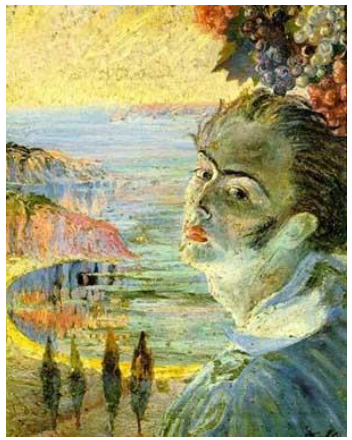
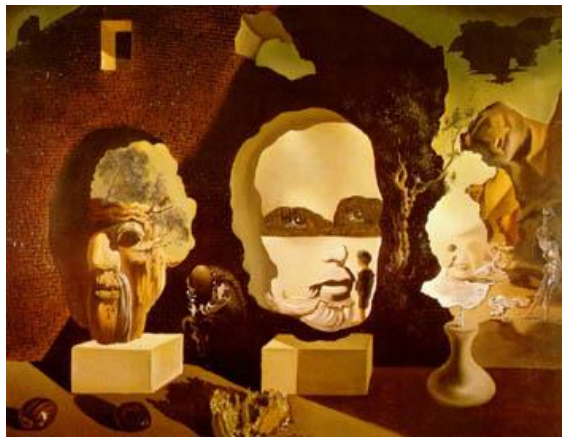
It is still early days for assessment. So far, however, trainees have repeatedly emphasized that gaining a broader perspective has been helpful. In future, we will collect information about the impact that the R3 approach has on graduates' career choices and achievements.

We believe that researchers who are educated more broadly will do science more thoughtfully, with the result that other scientists, and society at large, will be able to rely on this work for a better, more rational world. Science should strive to be self-improving, not just self-correcting. ■

Gundula Bosch *directs the R3 Graduate Science Initiative at Johns Hopkins Bloomberg School of Public Health in Baltimore, Maryland.*
e-mail: gbosch@jhu.edu

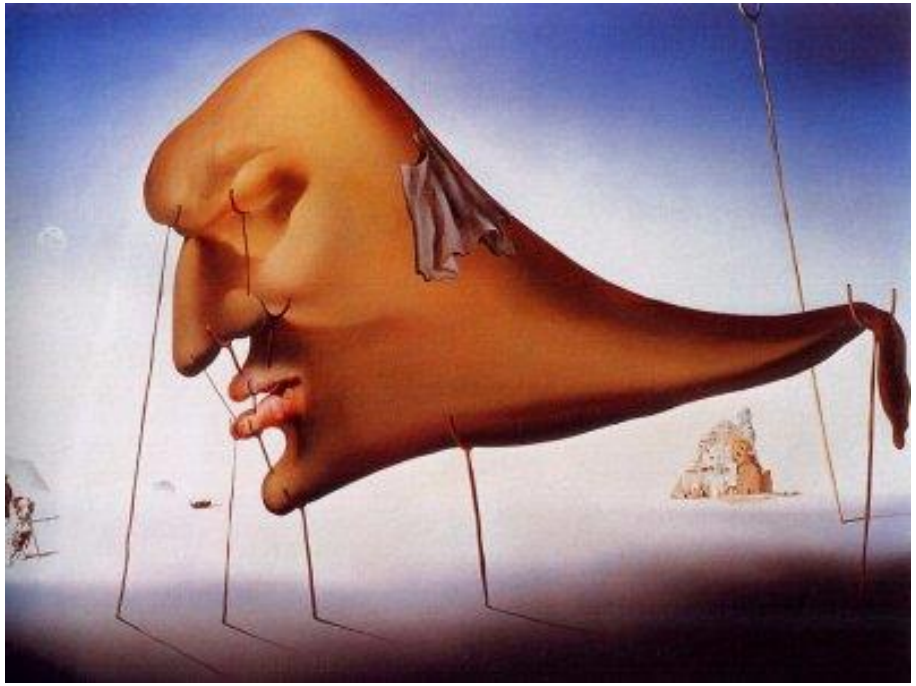
Paintings by two different painters



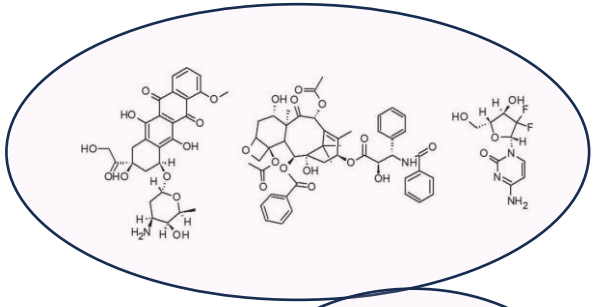





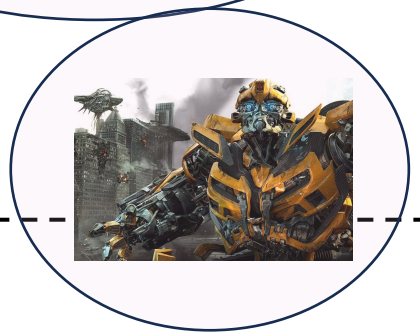
Who’s painting is this?

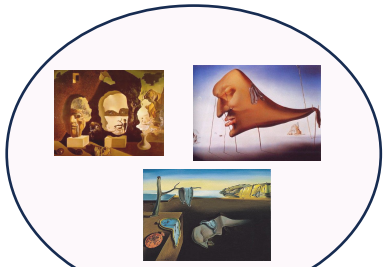
And this?

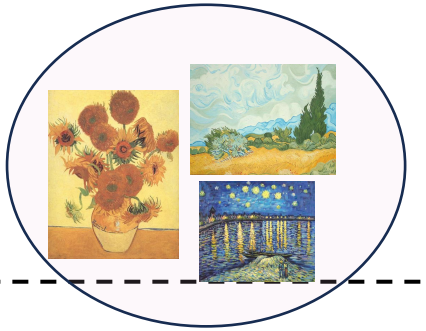


learning from data for generalization to unseen cases

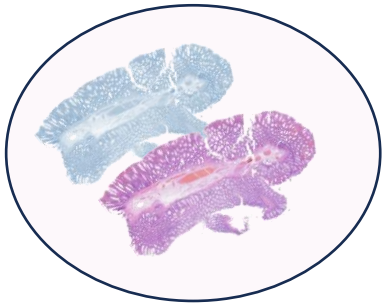


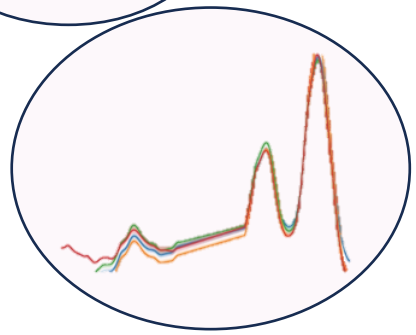


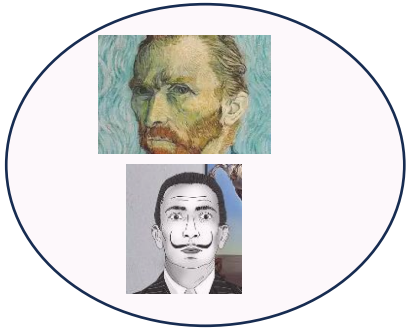




A transformer that can transform into a yellow car







I. Entities have (explicit or implicit) representations

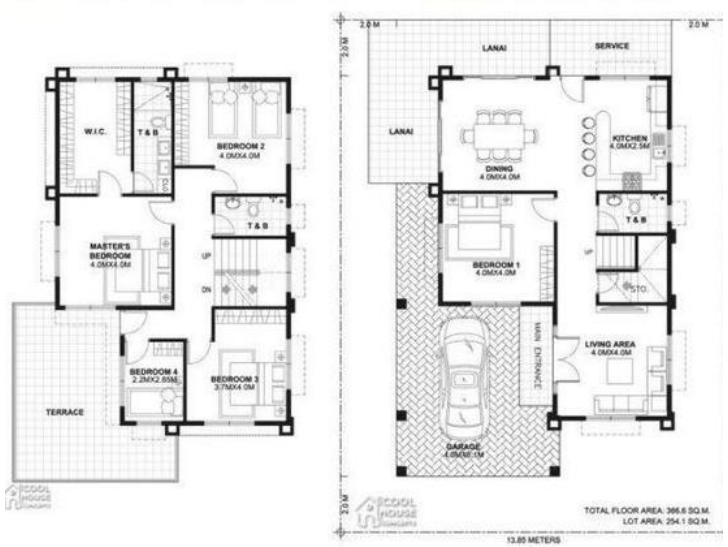
"Bank" in which statement is more semantically related to the picture?

- **A:** As he walked by the **bank**, he saw some boats
- **B:** As he walked by the **bank**, he saw some tellers

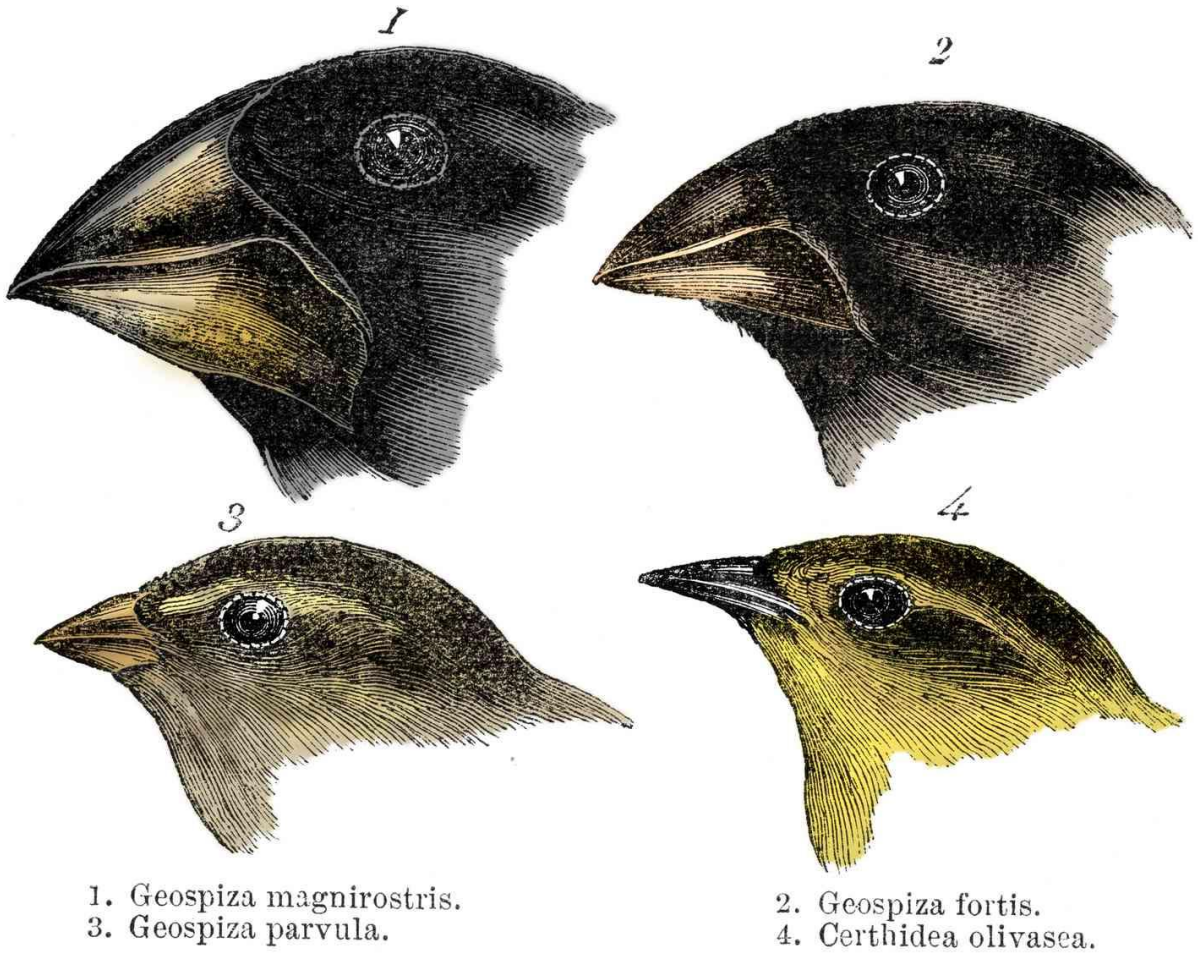
As he walked by the **bank**, he saw some boats

As he walked by the **bank**, he saw some tellers

II. Semantic relatedness of entities is context dependent and thus their representations are contextual



III. Representation of any entity can allow us to reconstruct or “generate” it



1. Geospiza magnirostris.
3. Geospiza parvula.

2. Geospiza fortis.
4. Certhidea olivasea.

IV. It is possible to develop representations in an inductive manner (through empirical observations)

US Airways Flight 1549

V. Intelligence is the capacity to develop and utilize causal representations of entities, enabling an organism or system to act effectively and adaptively.

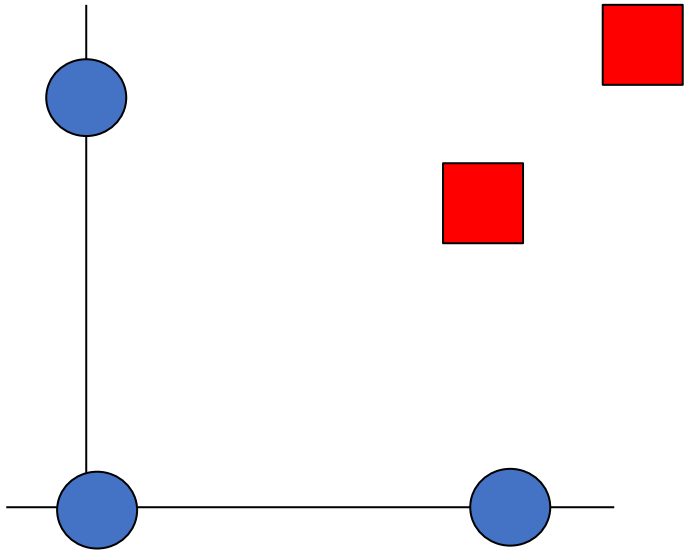
Only if we could have a mechanism that would enable developing such representations from empirical observations

Deep Learning

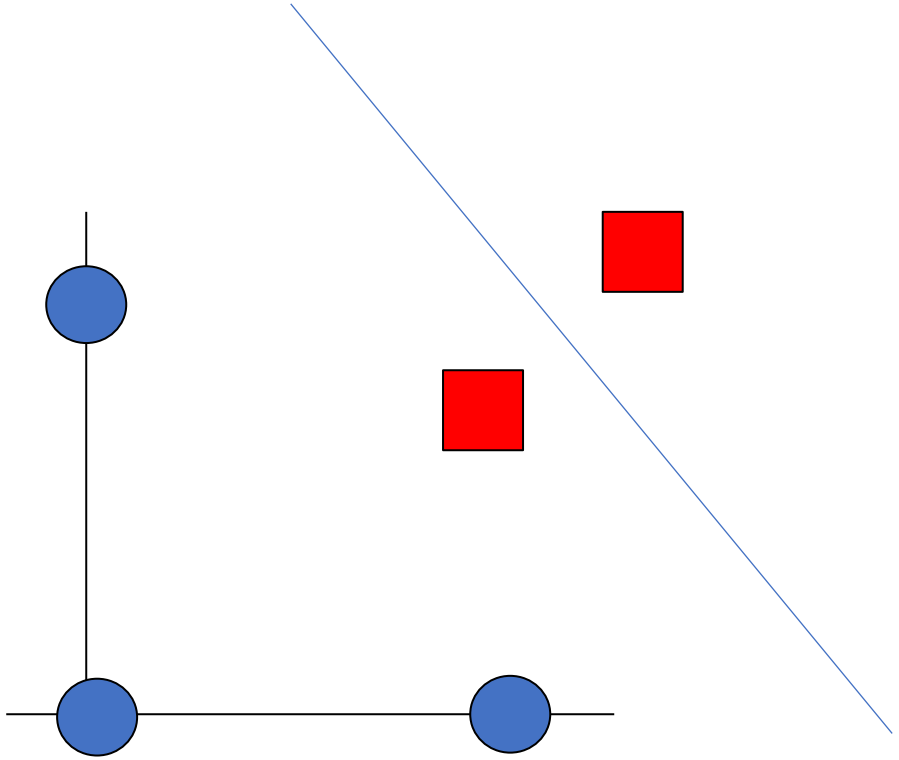
Learning Representations from training examples with “layers” of biologically inspired neurons

Working Principles

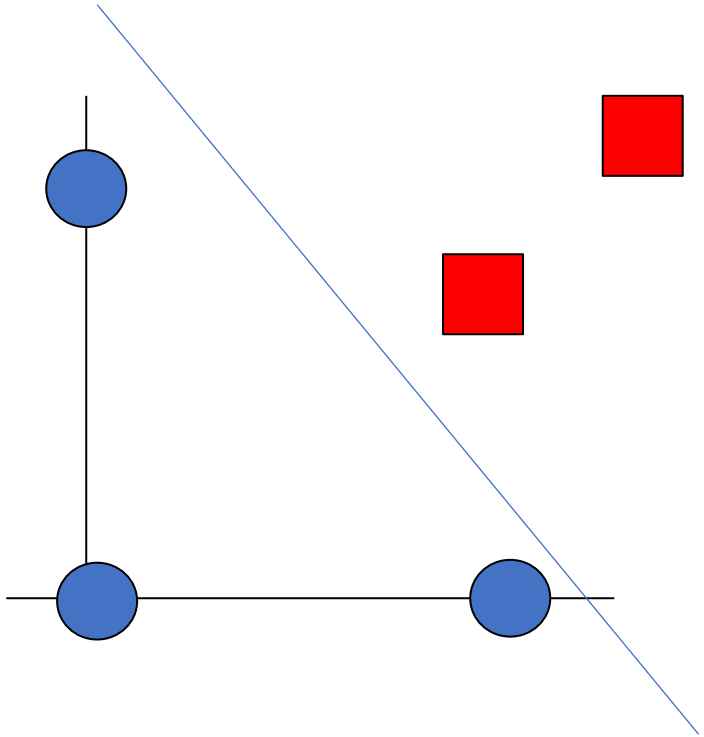
Exercise



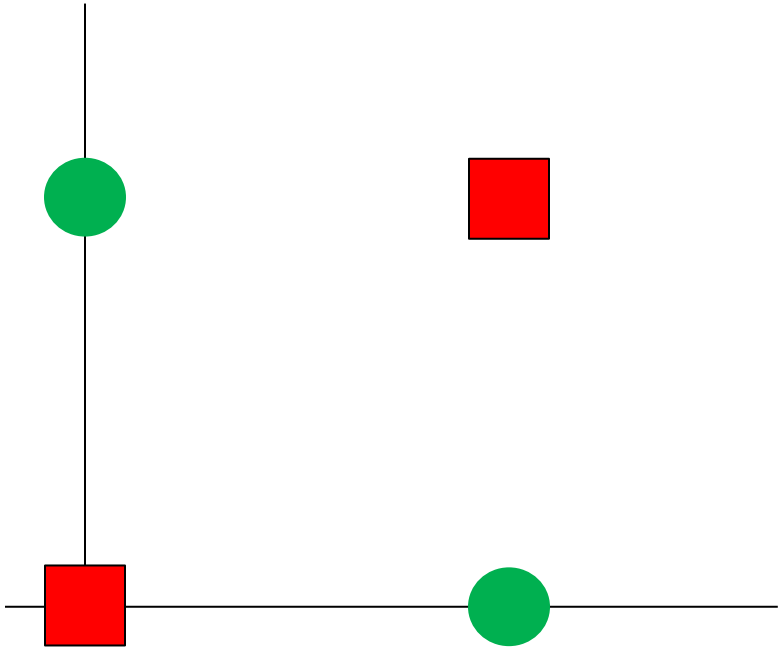
Exercise



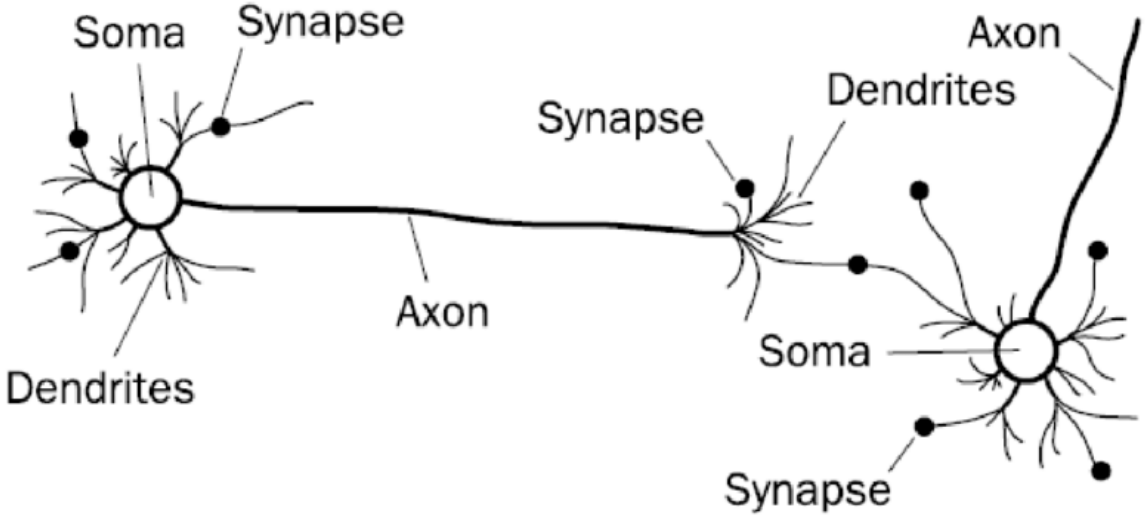
Exercise



But what if we have a linearly inseparable problems?

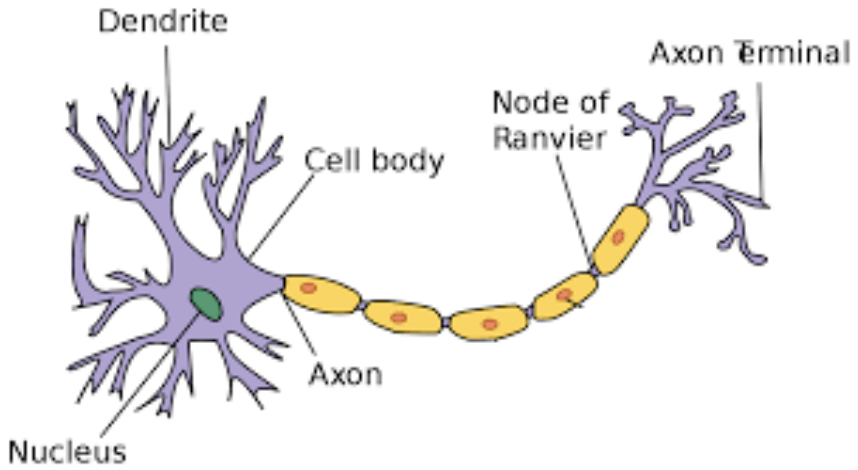


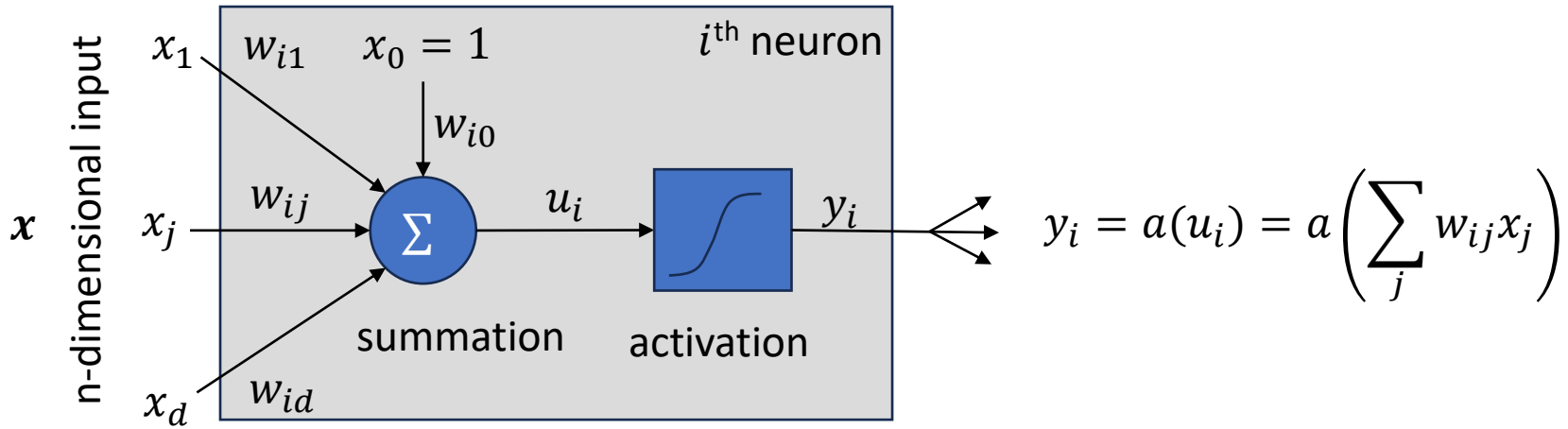
Biological Neurons and Networks



Single Neuron: Representation

- An abstraction of the biological neuron





$$y_i = a(u_i) = a\left(\sum_j w_{ij}x_j\right)$$

$$u_i = \sum_j w_{ij}x_j = \boldsymbol{w_i^T x}$$

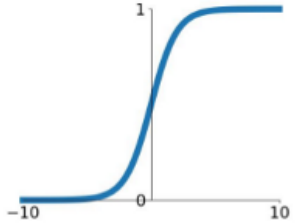
Activation Functions

Can use any activation function

Activation Functions

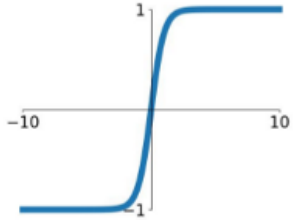
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



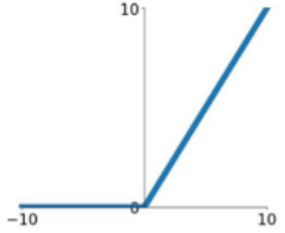
tanh

$$\tanh(x)$$



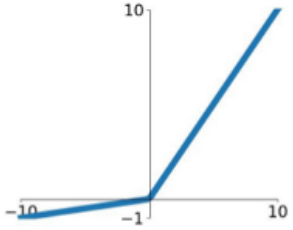
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

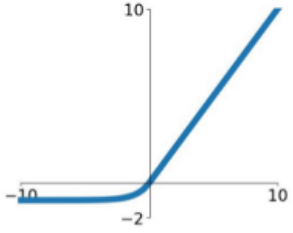


Maxout

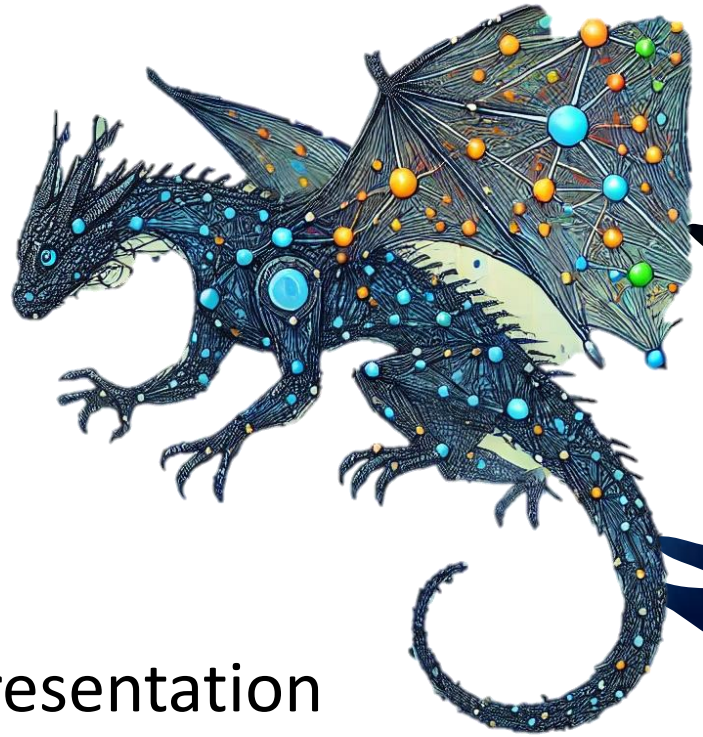
$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$

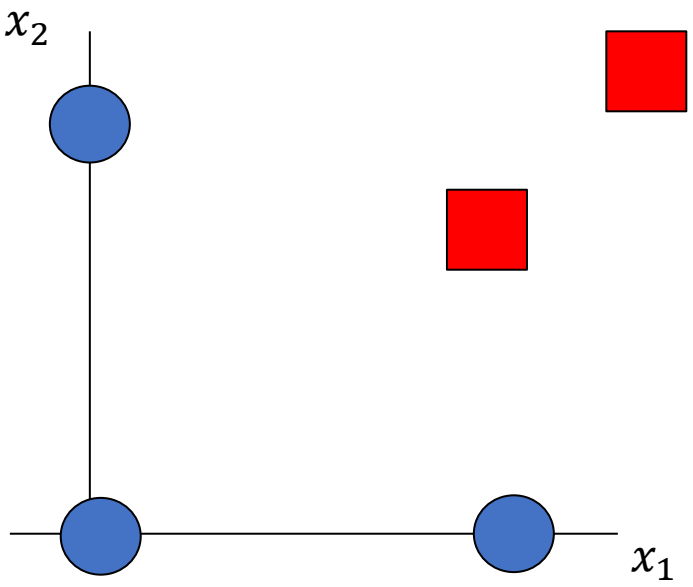


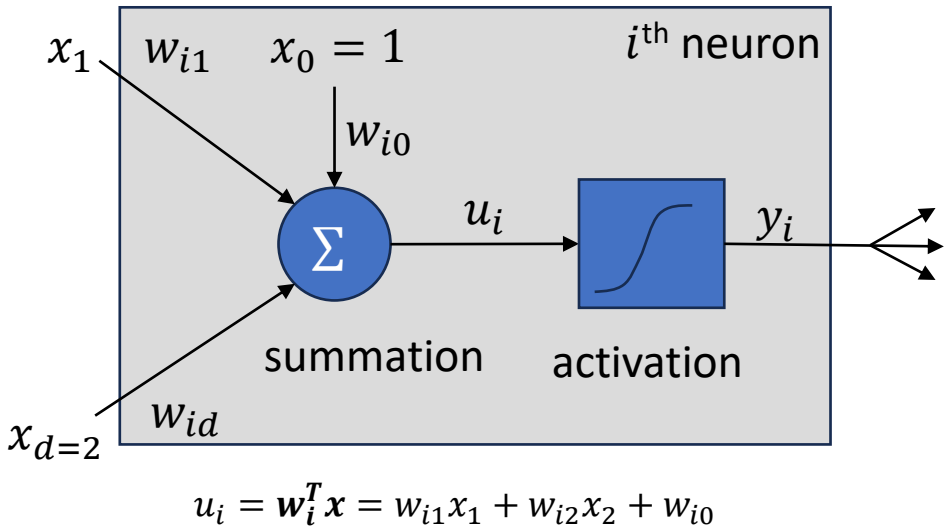
How to train your neural network



- **R**epresentation
- **E**valuation
- **O**ptimization

Representation: How does the model produce its output?





$$u_i = \boldsymbol{w_i^T x} = w_{i1}x_1 + w_{i2}x_2 + w_{i0}$$

$$y_i = a(u_i) = a\left(\sum_j w_{ij}x_j\right)$$

$$= a(w_{i1}x_1 + w_{i2}x_2 + w_{i0})$$

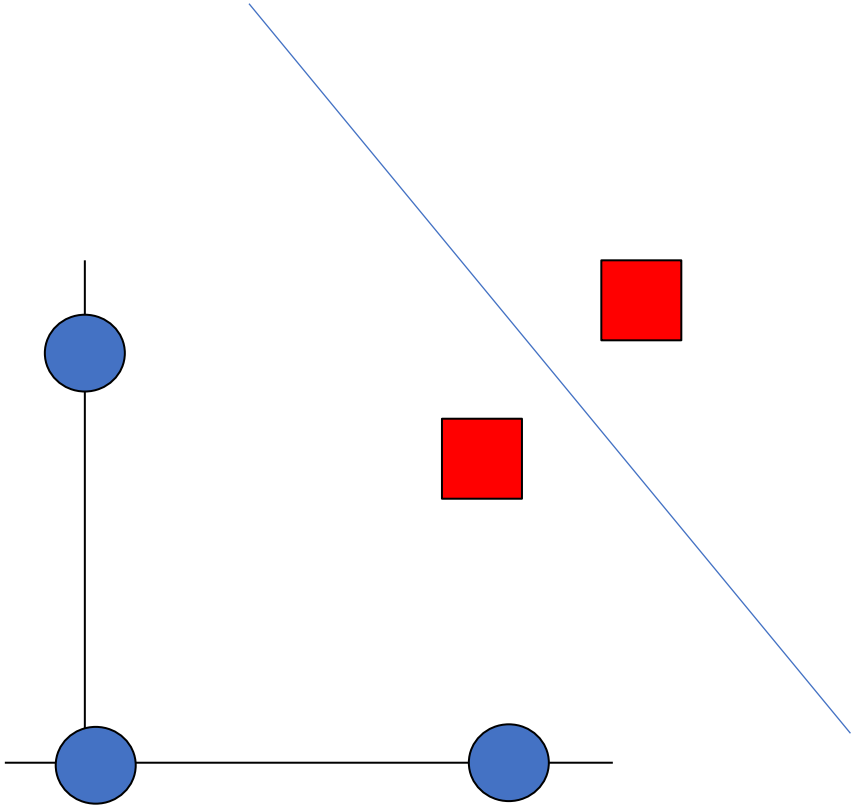
$$= w_{i1}x_1 + w_{i2}x_2 + w_{i0}$$

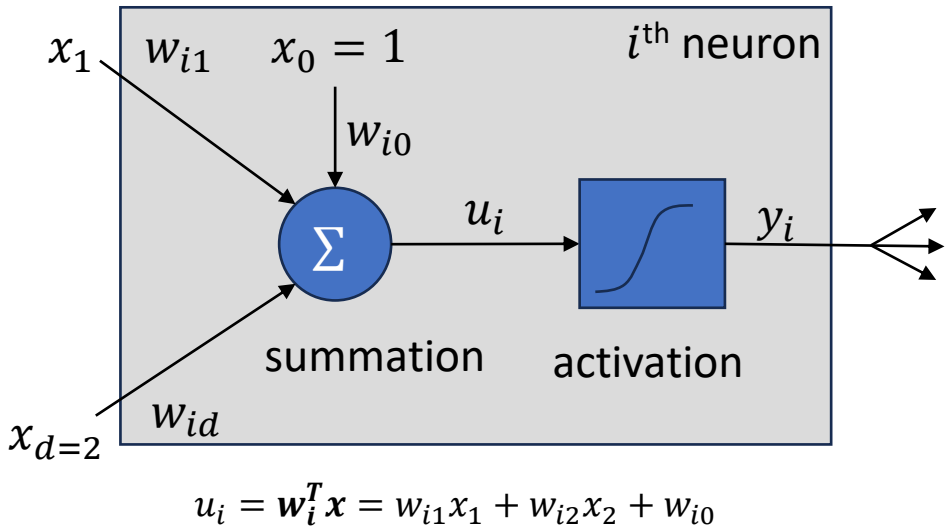
What do we want?

$$\boldsymbol{w_{i1}x_1 + w_{i2}x_2 + w_{i0} > 0}$$

$$\boldsymbol{w_{i1}x_1 + w_{i2}x_2 + w_{i0} < 0}$$

Evaluation



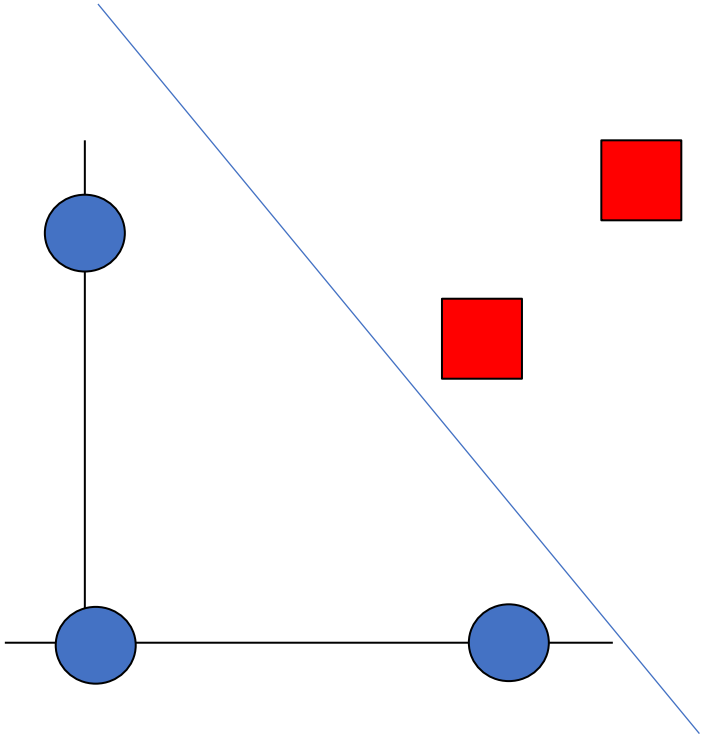


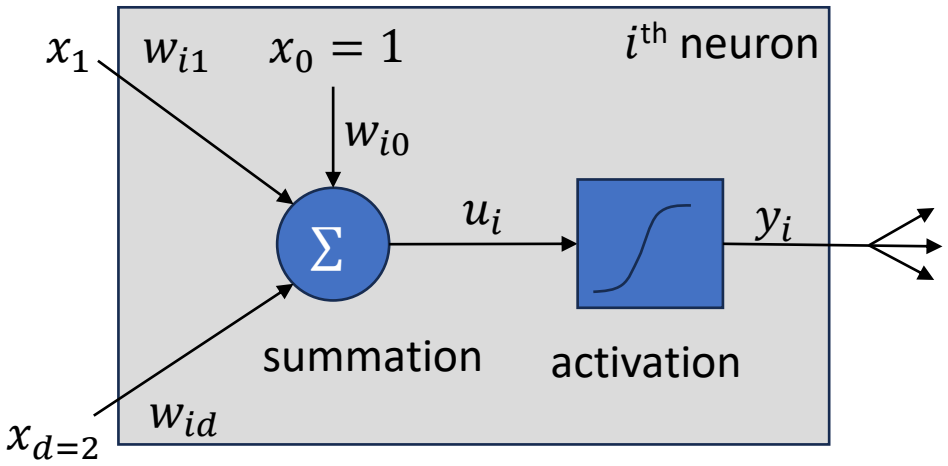
$$u_i = \boldsymbol{w_i^T x} = w_{i1}x_1 + w_{i2}x_2 + w_{i0}$$

$$y_i = a(u_i) = a\left(\sum_j w_{ij}x_j\right)$$
$$= a(w_{i1}x_1 + w_{i2}x_2 + w_{i0})$$
$$= w_{i1}x_1 + w_{i2}x_2 + w_{i0}$$

Calculate how much error the line produces

Optimization





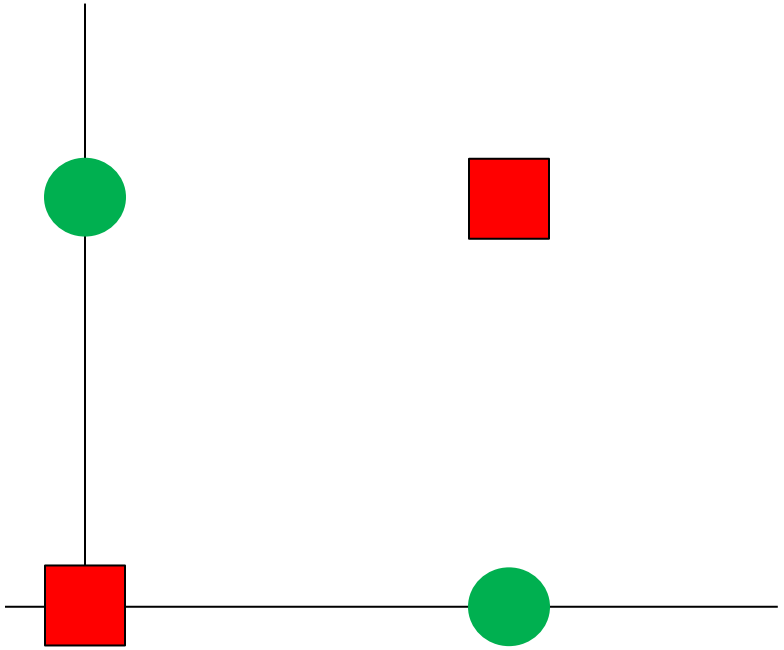
$$u_i = \boldsymbol{w_i^T x} = w_{i1}x_1 + w_{i2}x_2 + w_{i0}$$

$$y_i = a(u_i) = a\left(\sum_j w_{ij}x_j\right)$$

$$= a(w_{i1}x_1 + w_{i2}x_2 + w_{i0})$$

$$= w_{i1}x_1 + w_{i2}x_2 + w_{i0}$$

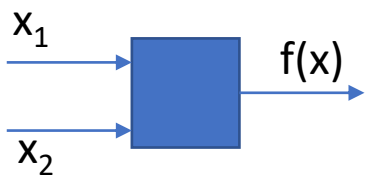
Update Weight parameters to reduce error (using gradient descent)



How to fold the space?

- Change the definition of distance between points

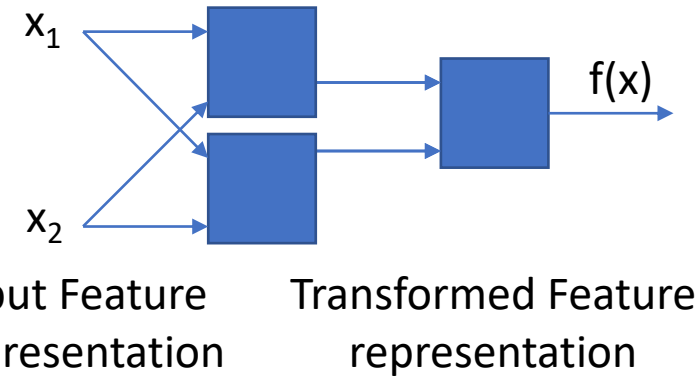
- How to achieve this?
  - By transforming the features of the examples to another space



How to fold the space?

- Change the definition of distance between points

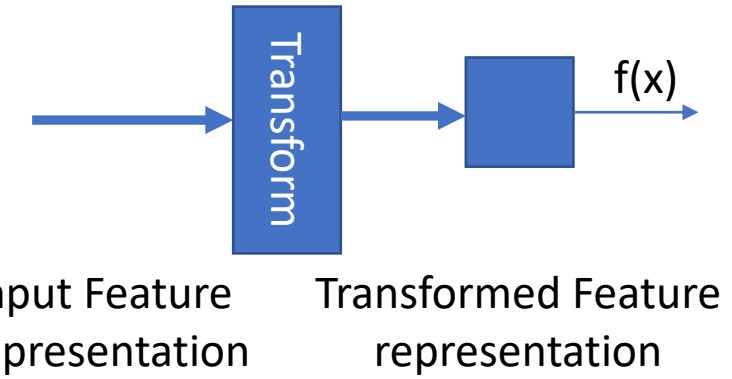
- How to achieve this?
  - By transforming the features of the examples to another space

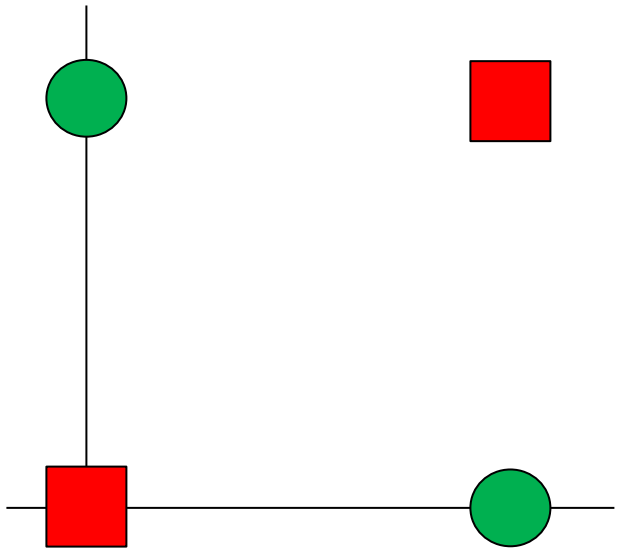


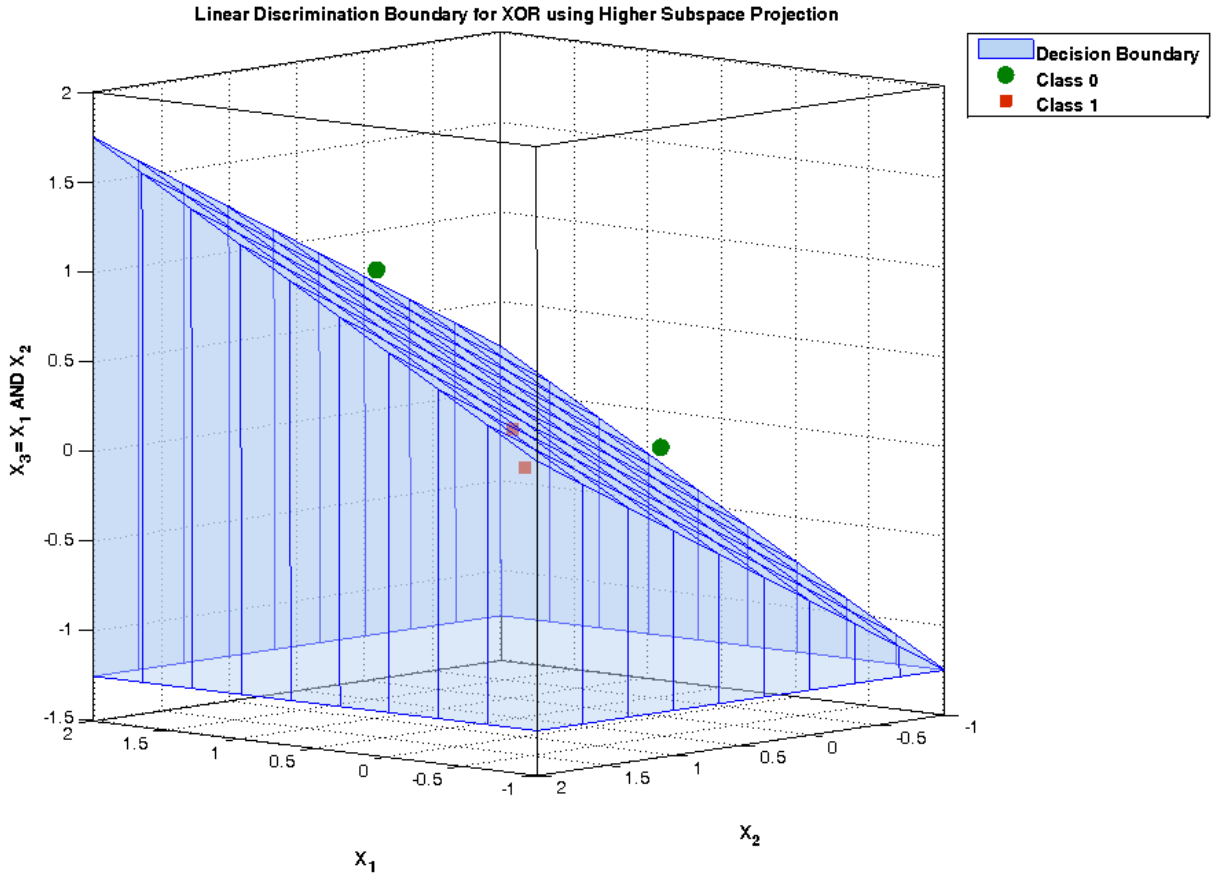
How to fold the space?

- Change the definition of distance between points

- How to achieve this?
  - By transforming the features of the examples to another space
  - At an abstract level







$x^{(1)}$	$x^{(2)}$	y
0	0	-1
0	1	+1
1	0	+1
1	1	-1

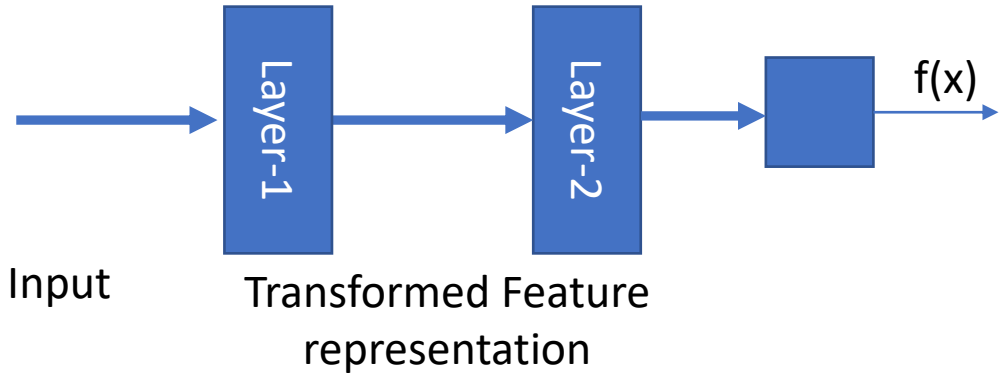
$$\boldsymbol{\phi}\left(\begin{bmatrix} x^{(1)} \\ x^{(2)} \end{bmatrix}\right) = \begin{bmatrix} x^{(1)^2} \\ x^{(2)^2} \\ \sqrt{2}x^{(1)}x^{(2)} \end{bmatrix}$$

$x'^{(1)}$	$x'^{(2)}$	$x'^{(3)}$	y
0	0	0	-1
0	1	0	+1
1	0	0	+1
1	1	$\sqrt{2}$	-1

How to fold the space?

- Change the definition of distance between points

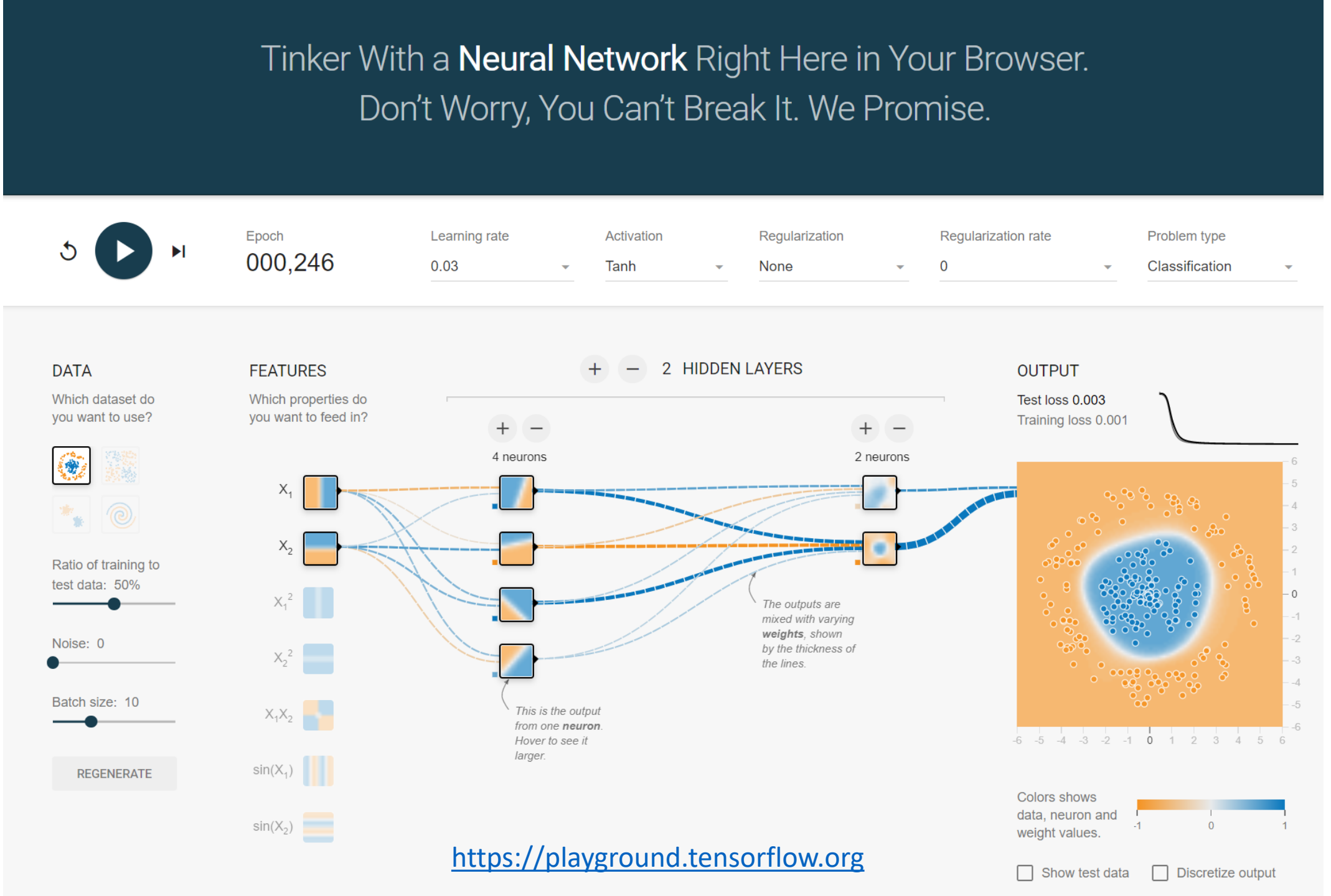
- How to achieve this?
  - By transforming the features of the examples to another space
  - We can do it multiple times



Why deep learning?

- By multiple layers of neurons, we can achieve
  - Drawing lines
  - Linear separability
  - Implicit representation learning

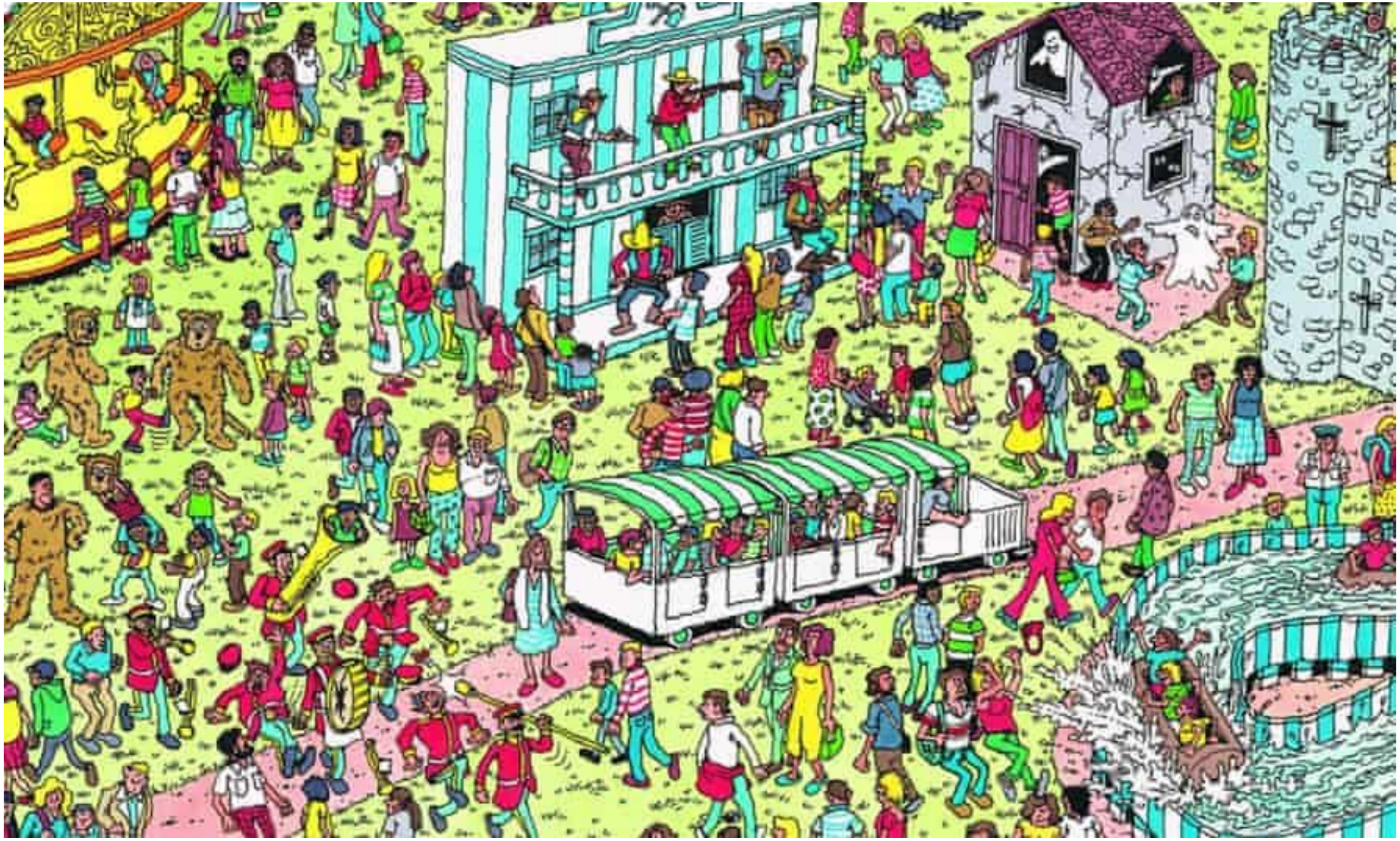
- Deeper architectures are more “efficient at learning representations”
  - We need fewer cuts to cut a shape if we fold many times

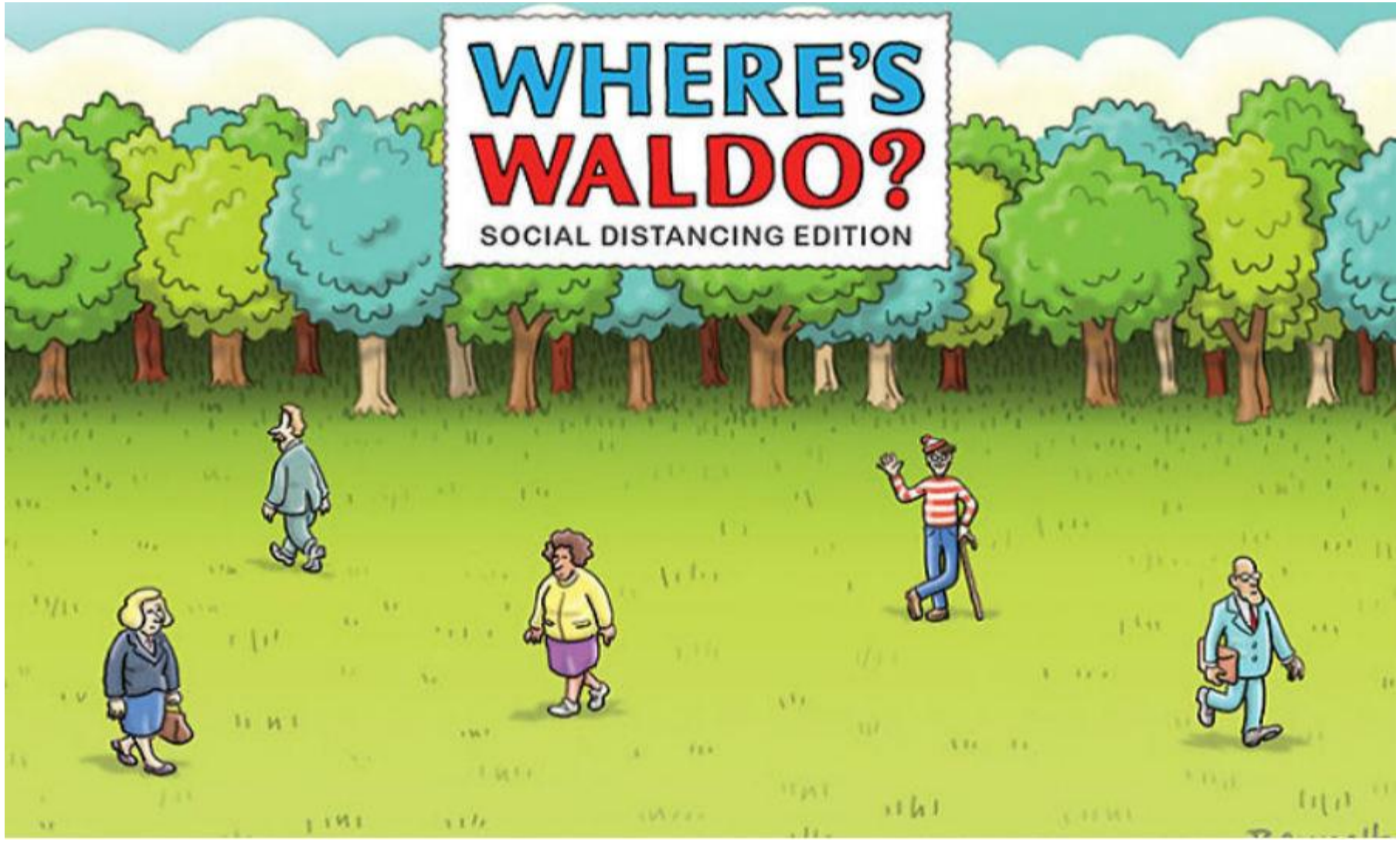


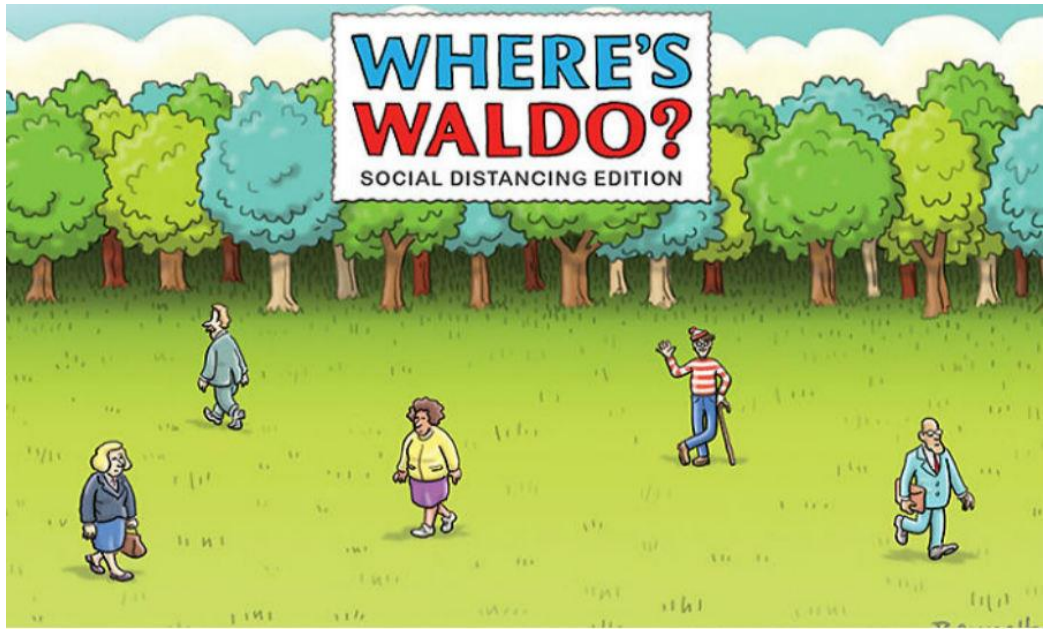
https://playground.tensorflow.org

Convolutional Neural Networks

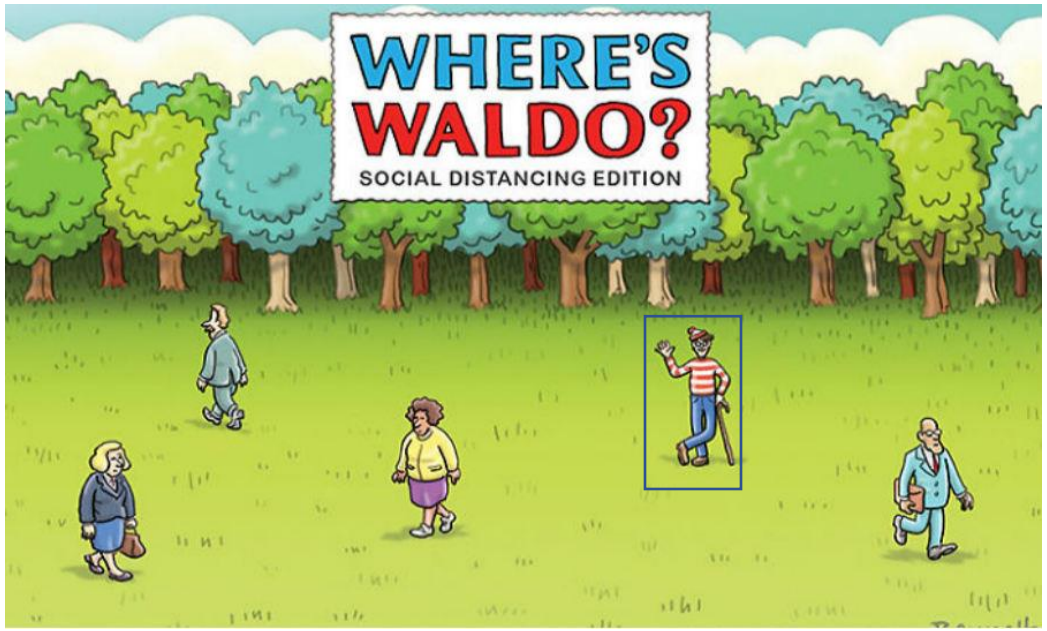
Where’s Waldo?

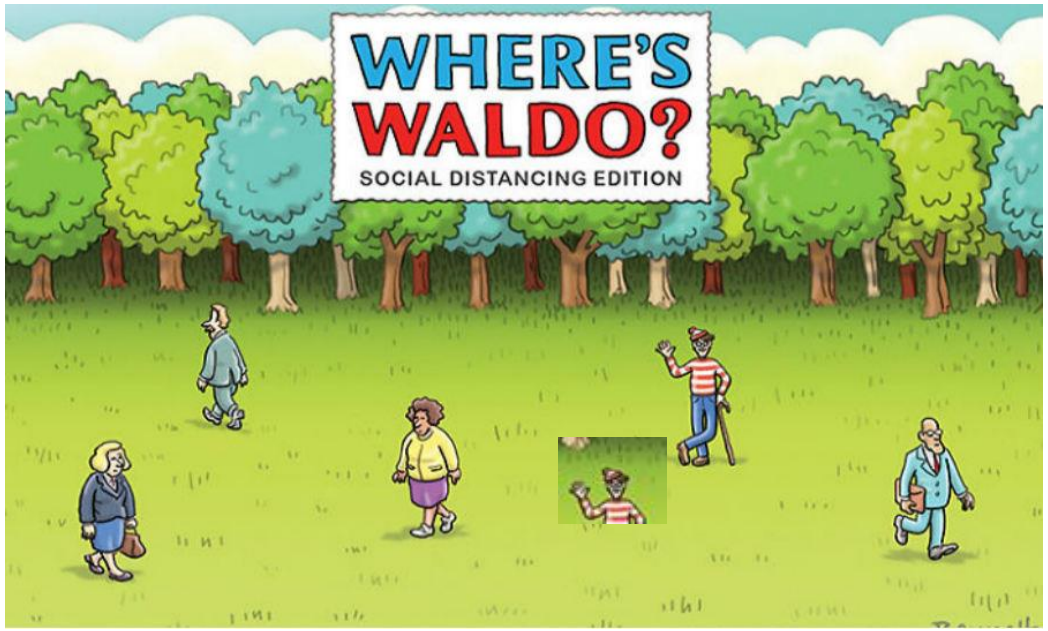



WHERE'S
WALDO?
SOCIAL DISTANCING EDITION


WHERE'S
WALDO?
SOCIAL DISTANCING EDITION

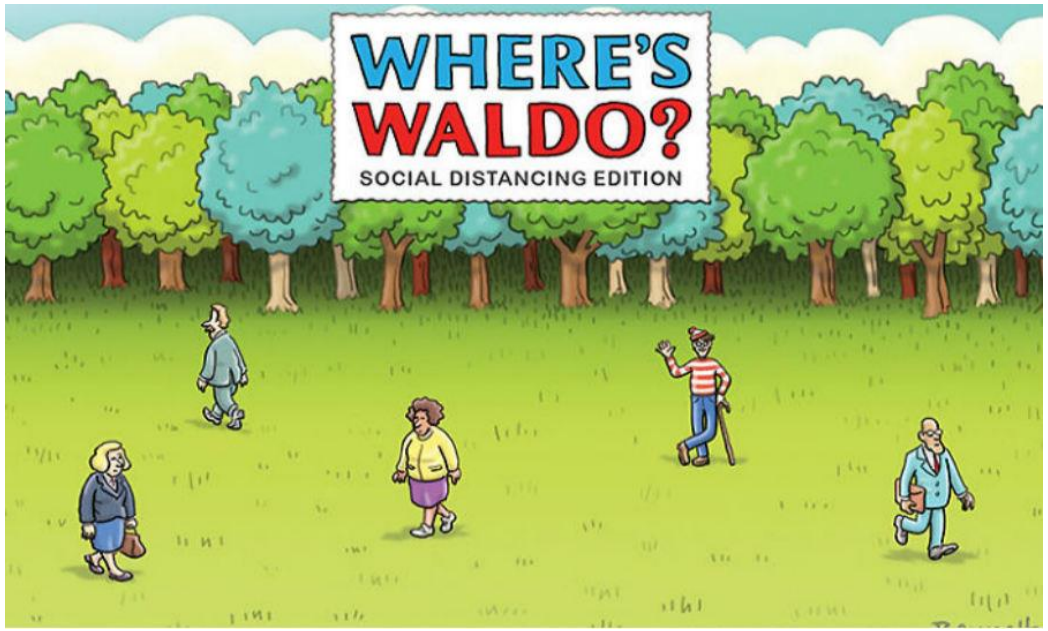
Model


WHERE'S
WALDO?
SOCIAL DISTANCING EDITION



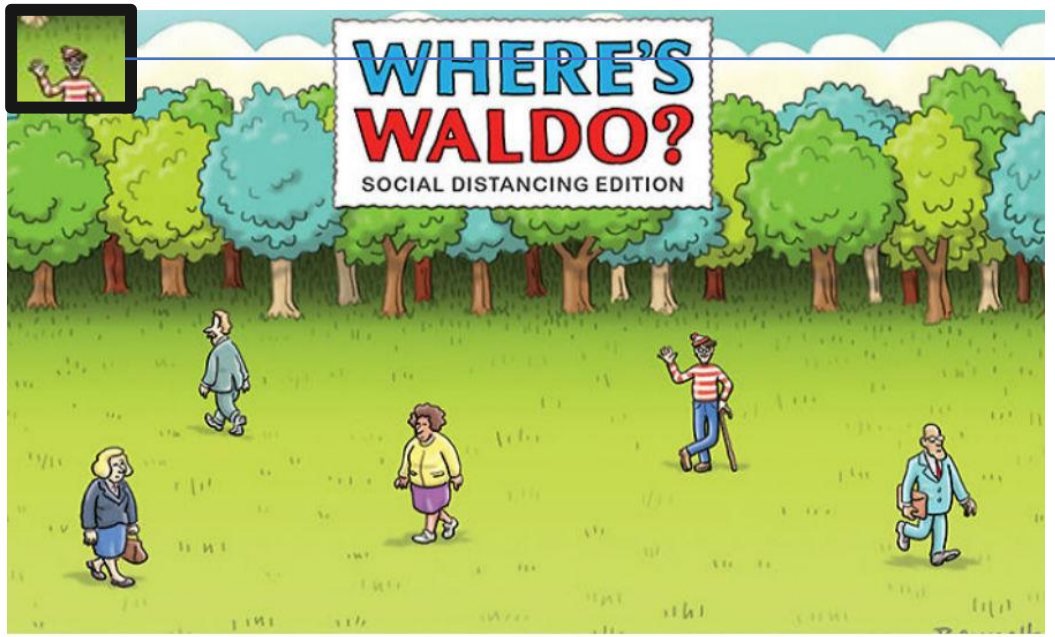
Model

- Assume we have a cutout of what Waldo looks like
- And if we “scan” (formally called correlate or convolve) the cutout against the input image – we should see a peak at the location where Waldo occurs in the input image



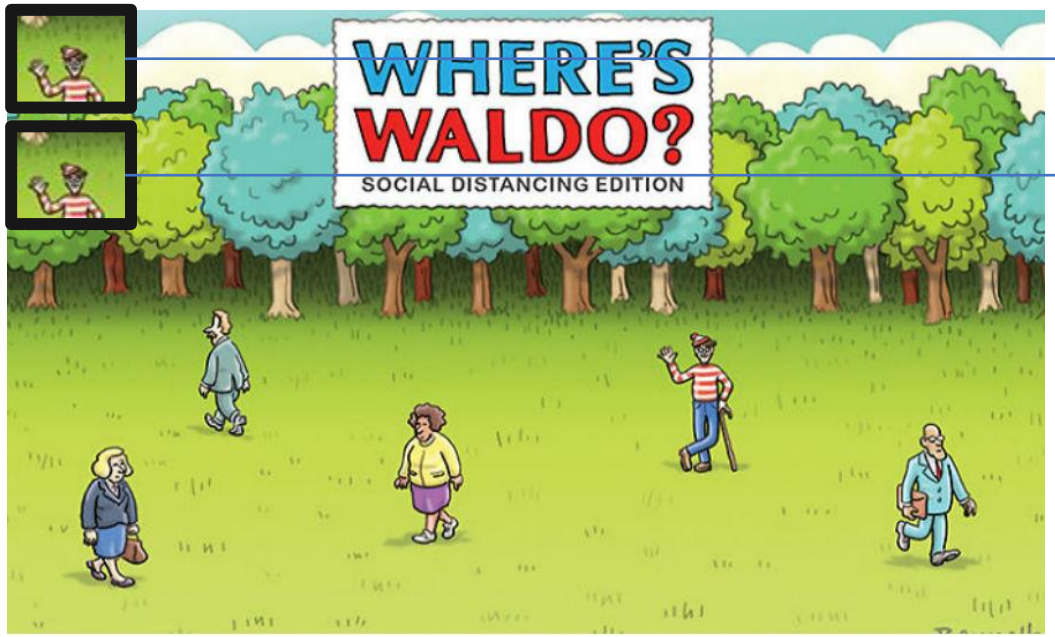
Model

- Assume we have a cutout of what Waldo looks like
- And if we “scan” (formally called correlate or convolve) the cutout against the input image – we should see a peak at the location where Waldo occurs in the input image



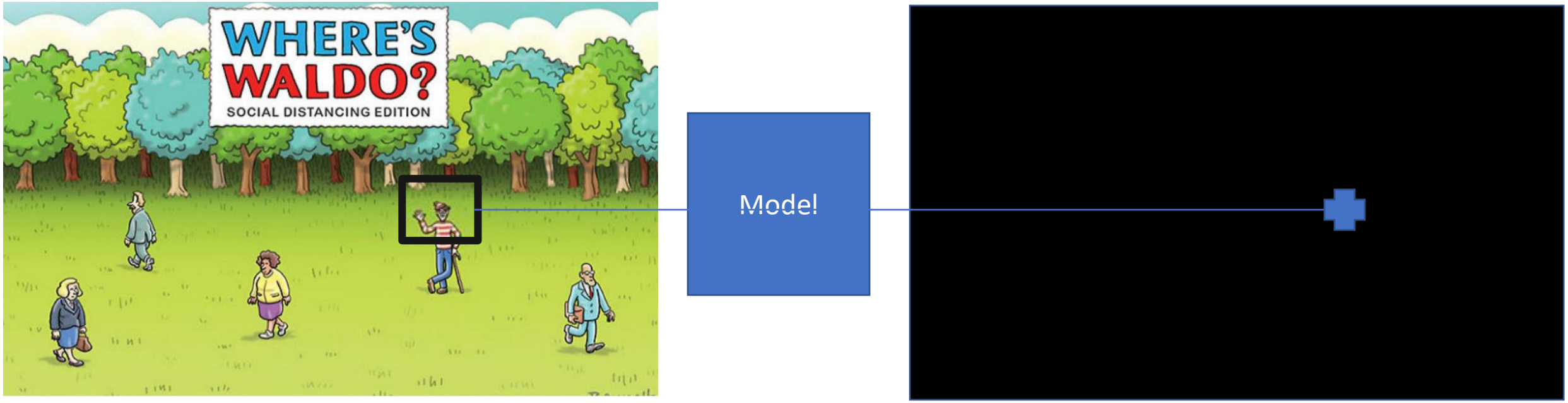
Model

- Assume we have a cutout of what Waldo looks like
- And if we “scan” (formally called correlate or convolve) the cutout against the input image – we should see a peak at the location where Waldo occurs in the input image



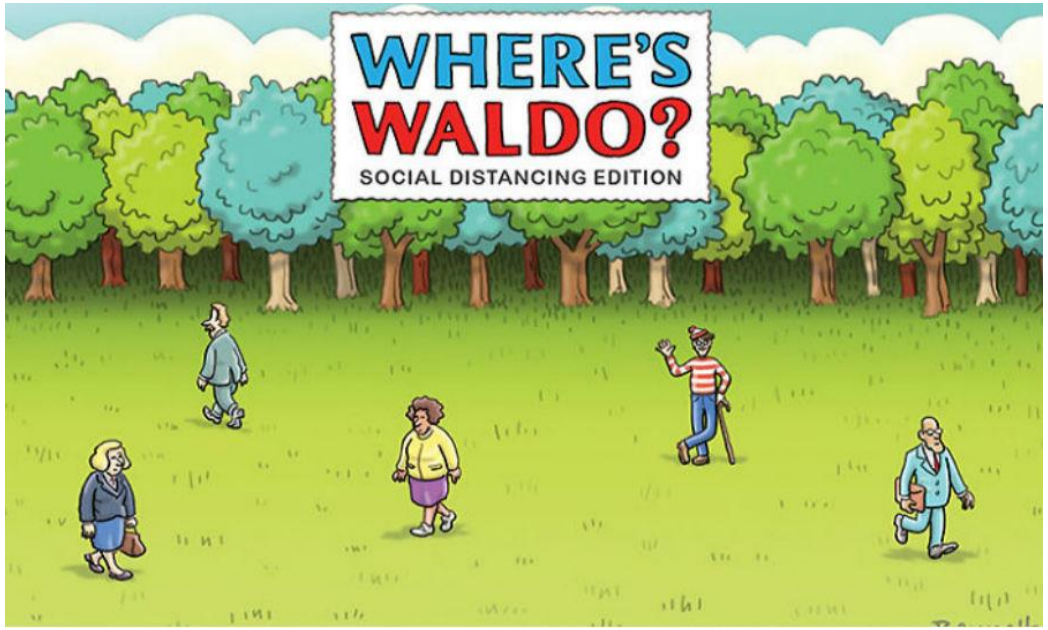
Model

- Assume we have a cutout of what Waldo looks like
- And if we “scan” (formally called correlate or convolve) the cutout against the input image – we should see a peak at the location where Waldo occurs in the input image

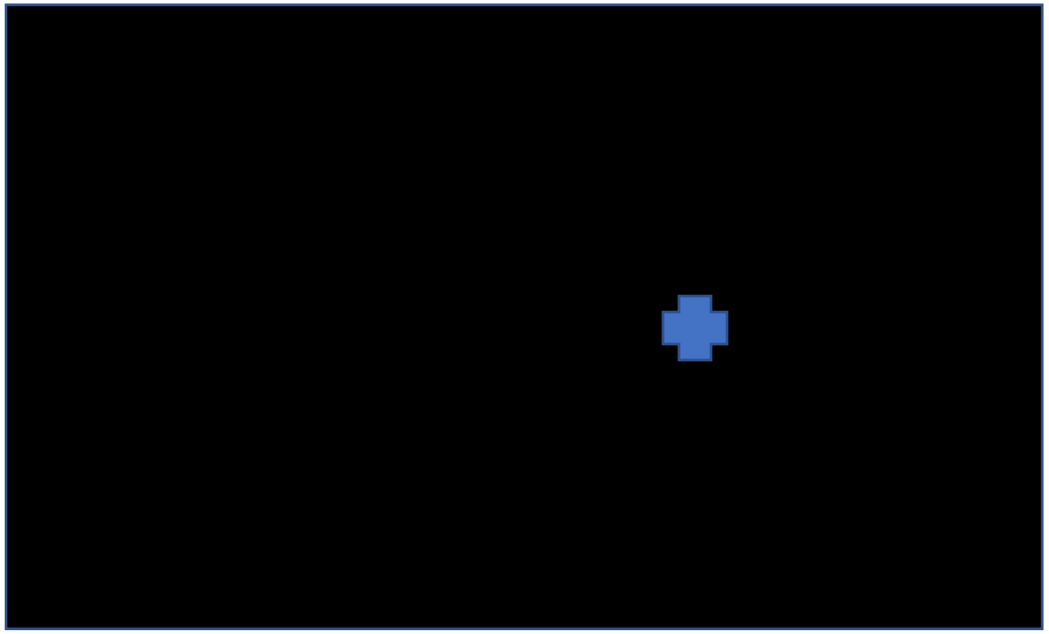


- What if I don’t have a cut out of Waldo?

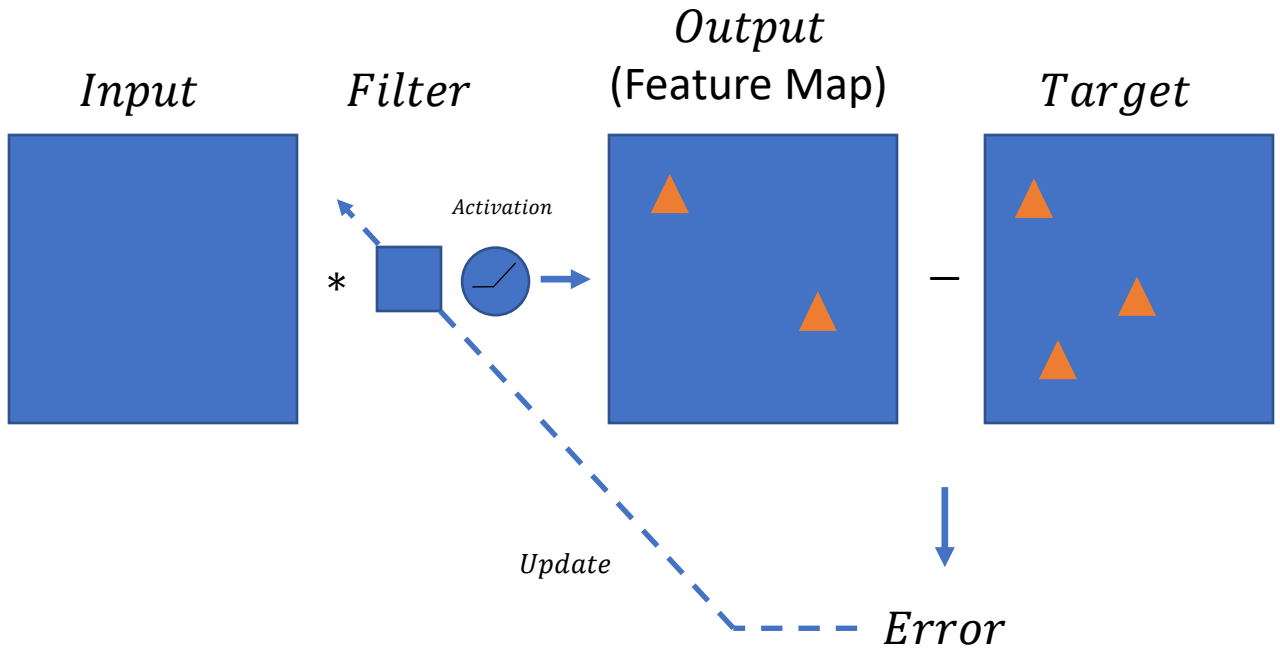
- Can we find him still?



Model



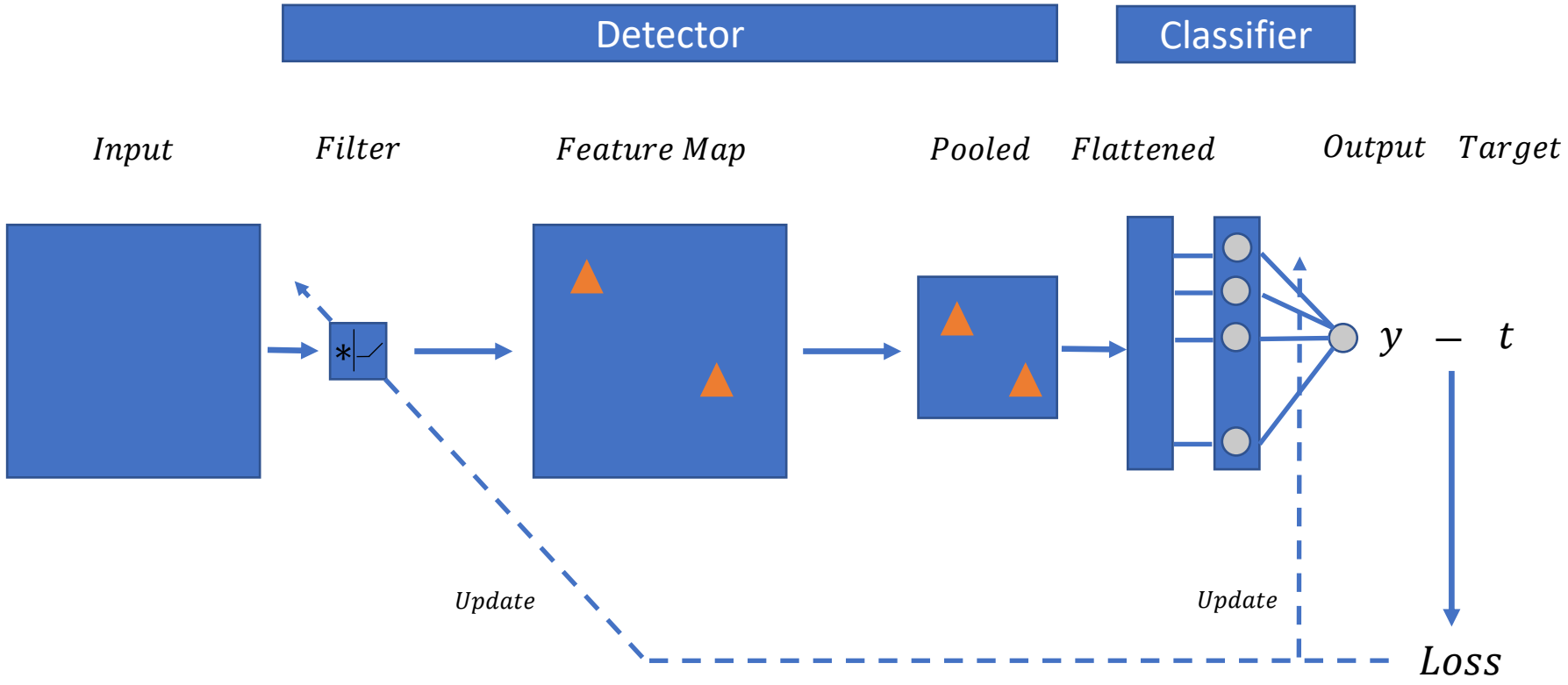
(Very) basic "convolutional neural network"



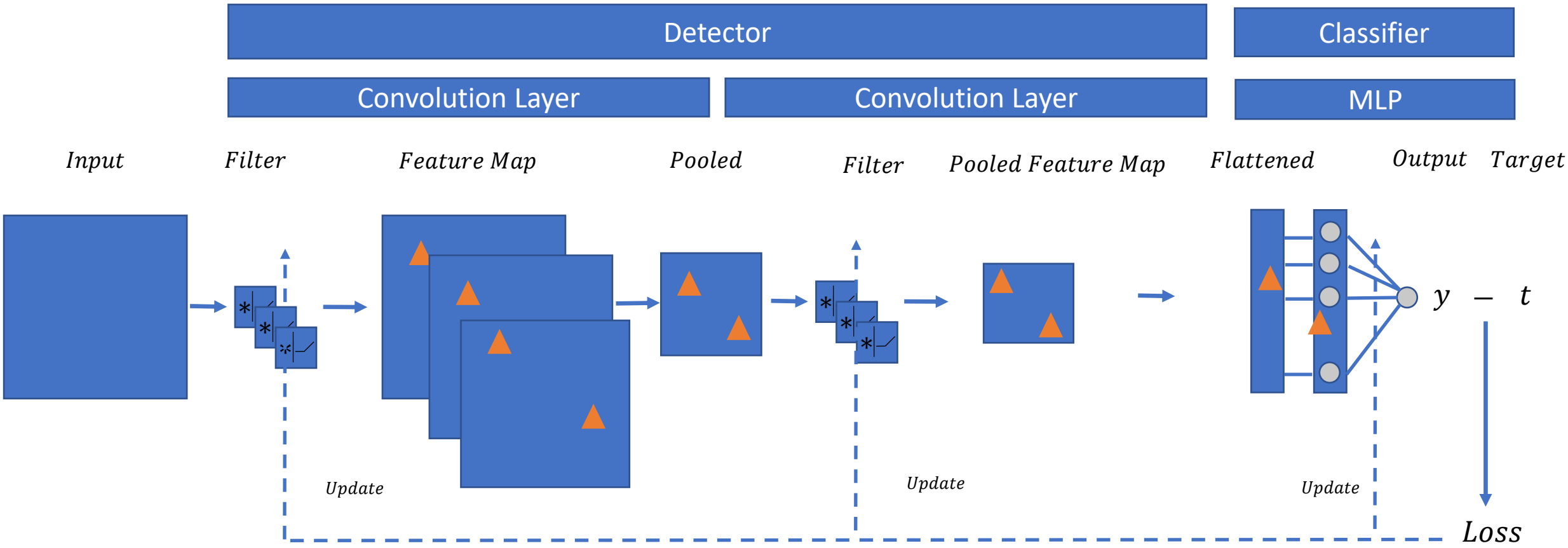
- Acts as a "detection" or "feature extraction" unit

https://github.com/foxtrotmike/CS909/blob/master/learn_filters.ipynb
https://github.com/foxtrotmike/CS909/blob/master/cnn_mnist_pytorch.ipynb

Basic convolutional neural network for ML



CNNs



https://poloclub.github.io/cnn-explainer/

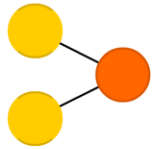
A mostly complete chart of

Neural Networks

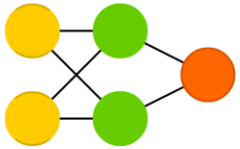
http://www.asimovinstitute.org/neural-network-zoo/

- Backfed Input Cell
- Input Cell
- Noisy Input Cell
- Hidden Cell
- Probablistic Hidden Cell
- Spiking Hidden Cell
- Output Cell
- Match Input Output Cell
- Recurrent Cell
- Memory Cell
- Different Memory Cell
- Kernel
- Convolution or Pool

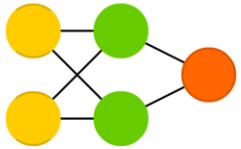
Perceptron (P)



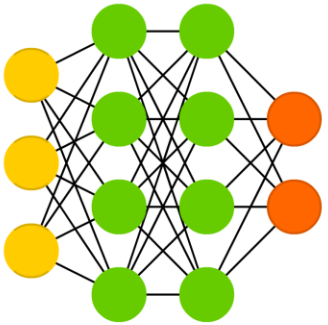
Feed Forward (FF)



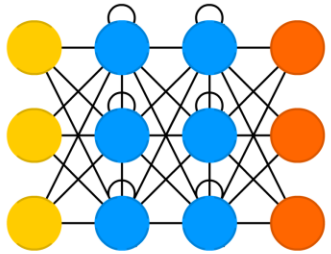
Radial Basis Network (RBF)



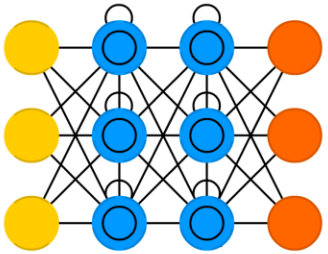
Deep Feed Forward (DFF)



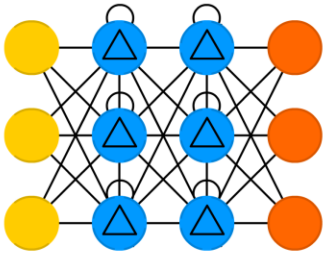
Recurrent Neural Network (RNN)



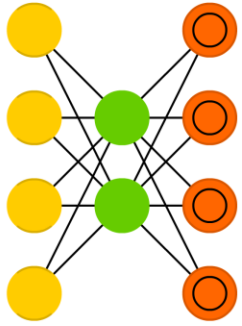
Long / Short Term Memory (LSTM)



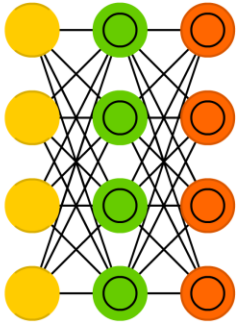
Gated Recurrent Unit (GRU)



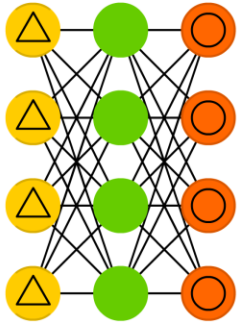
Auto Encoder (AE)



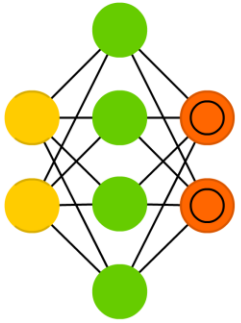
Variational AE (VAE)

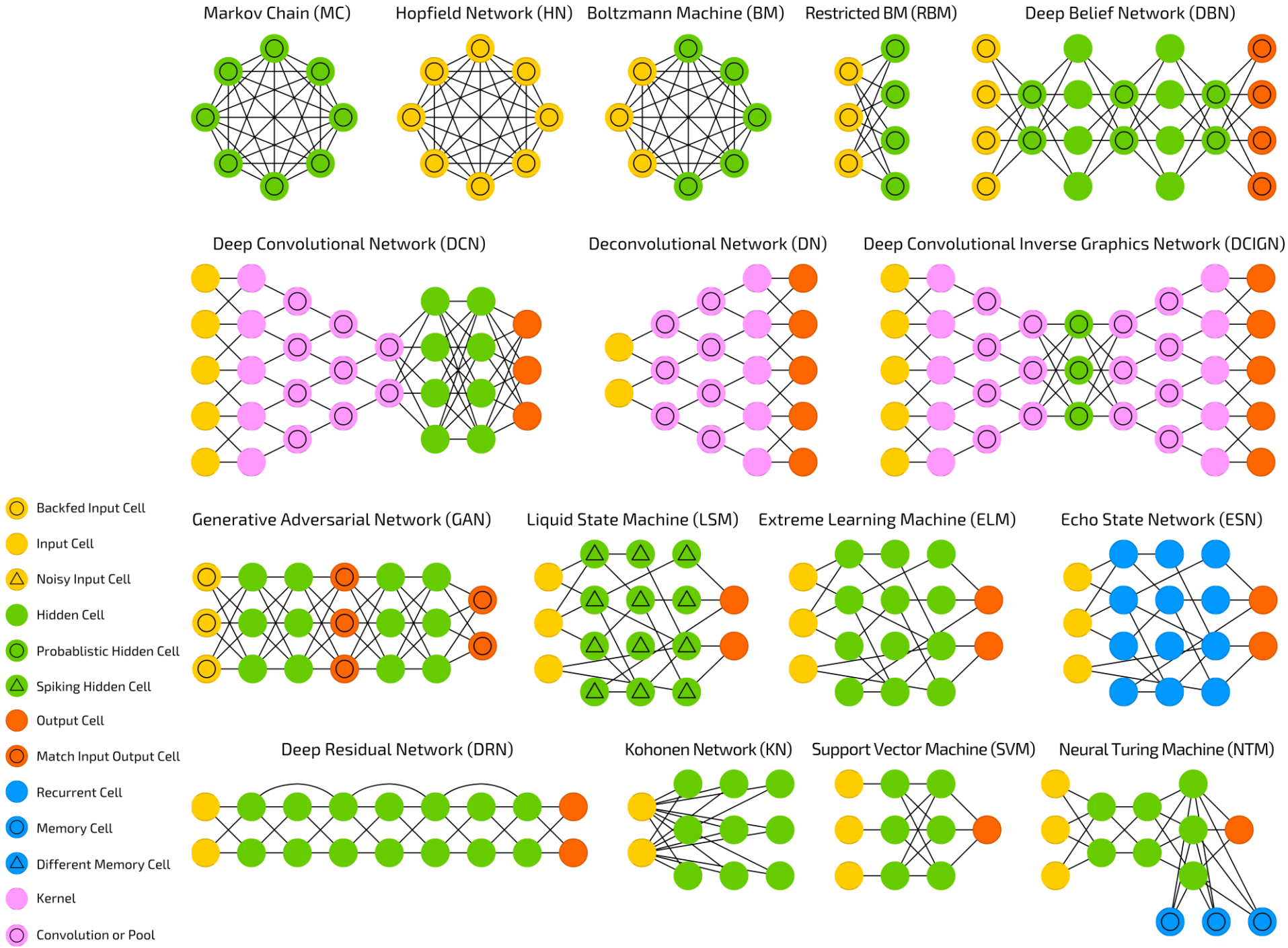


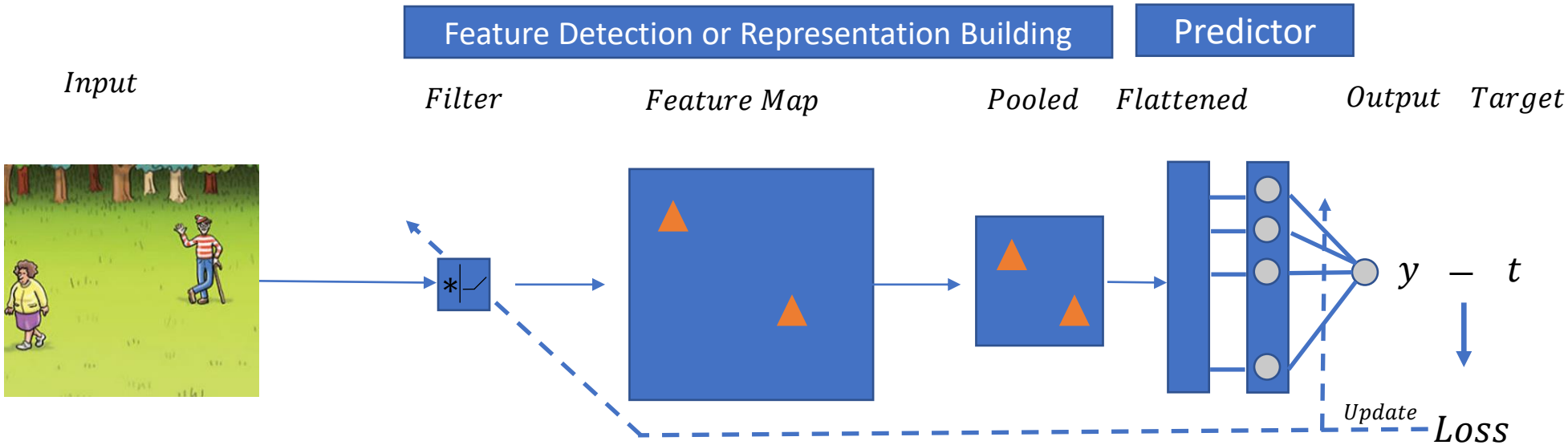
Denoising AE (DAE)



Sparse AE (SAE)

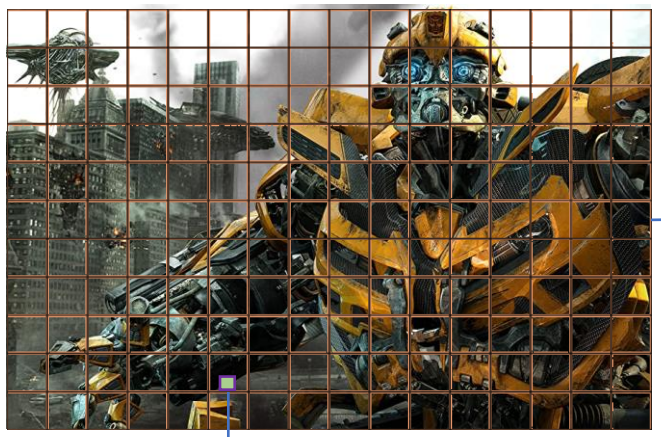



Markov Chain (MC)
Hopfield Network (HN)
Boltzmann Machine (BM)
Restricted BM (RBM)
Deep Belief Network (DBN)
Deep Convolutional Network (DCN)
Deconvolutional Network (DN)
Deep Convolutional Inverse Graphics Network (DCIGN)
Backfed Input Cell
Input Cell
Noisy Input Cell
Hidden Cell
Probablistic Hidden Cell
Spiking Hidden Cell
Output Cell
Match Input Output Cell
Recurrent Cell
Memory Cell
Different Memory Cell
Kernel
Convolution or Pool
Generative Adversarial Network (GAN)
Liquid State Machine (LSM)
Extreme Learning Machine (ELM)
Echo State Network (ESN)
Deep Residual Network (DRN)
Kohonen Network (KN)
Support Vector Machine (SVM)
Neural Turing Machine (NTM)



Convolutional Neural Network

(Vision) Transformers



Transform representation of each token based on its semantic relatedness or "attention" score with other tokens

Build Patch "Embedding" Representation

$S_x = \{\boldsymbol{x_i} \in R^d | i = 1 \ldots n\}$

Transformer Encoding via Attention Blocks

Predictor

Optimus Prime
Bumblebee
Jazz
⋮
Ironhide
Ratchet

Output

Optimus Prime
Bumblebee
Jazz
⋮
Ironhide
Ratchet

Target

Weight Update

Loss

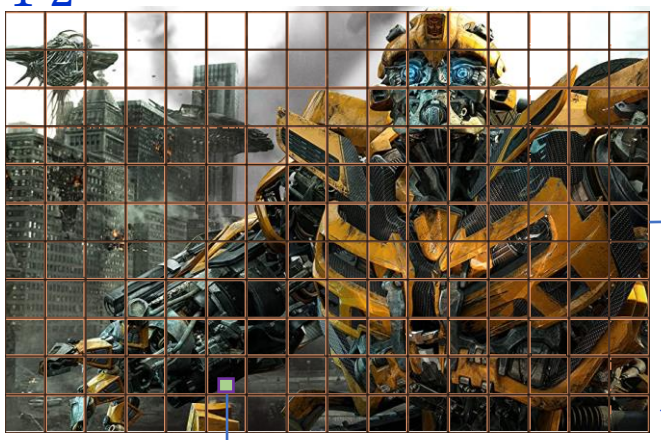
$\boldsymbol{x_i} \equiv \phi(\boldsymbol{f_i}, \boldsymbol{t_i})$

Feature Embedding: What is it?

Positional Embedding: Where is it?

Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, et al. "An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale." arXiv, June 3, 2021. https://doi.org/10.48550/arXiv.2010.11929.

(Vision) Transformers (for classification)



n tokens/patches

$\boldsymbol{x_i} \equiv \phi(\boldsymbol{f_i}, \boldsymbol{t_i})$ Feature Embedding: What is it?
Positional Embedding: Where is it?

Build Patch "Embedding" Representation
$S_x = \{\boldsymbol{x_i} \in R^d | i = 1 \ldots n\}$

Transformer Encoding via Attention Blocks

Predictor

Optimus
Bumblebee
Jazz
⋮
Ironhide
Ratchet

Optimus
Bumblebee
Jazz
⋮
Ironhide
Ratchet

Output — **Target**

$Loss$

$Weight\ Update$

Transform representation of each token based on its semantic relatedness or "attention" score with other tokens

A transformer that can transform into a yellow car is called _____.

$\boldsymbol{x_i} \equiv \phi(\boldsymbol{f_i}, \boldsymbol{t_i})$ Feature Embedding: What is it?
Positional Embedding: Where is it?

Build Token/Word "Embedding" Representation
$S_x = \{\boldsymbol{x_i} \in R^d | i = 1 \ldots n\}$

Transformer Encoding via Attention Blocks

Predictor

Optimus
Bumblebee
Jazz
⋮
Ironhide
Ratchet

Optimus
Bumblebee
Jazz
⋮
Ironhide
Ratchet

Output — **Target**

$Loss$

$Weight\ Update$

(NLP) Transformers (for next word prediction)

Simplest: $\phi(\boldsymbol{f_i}, \boldsymbol{t_i}) = \boldsymbol{f_i} + \boldsymbol{t_i}$

What is attention and why do you need it?

[Submitted on 12 Jun 2017 (v1), last revised 6 Dec 2017 (this version, v5)]

Attention Is All You Need

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin

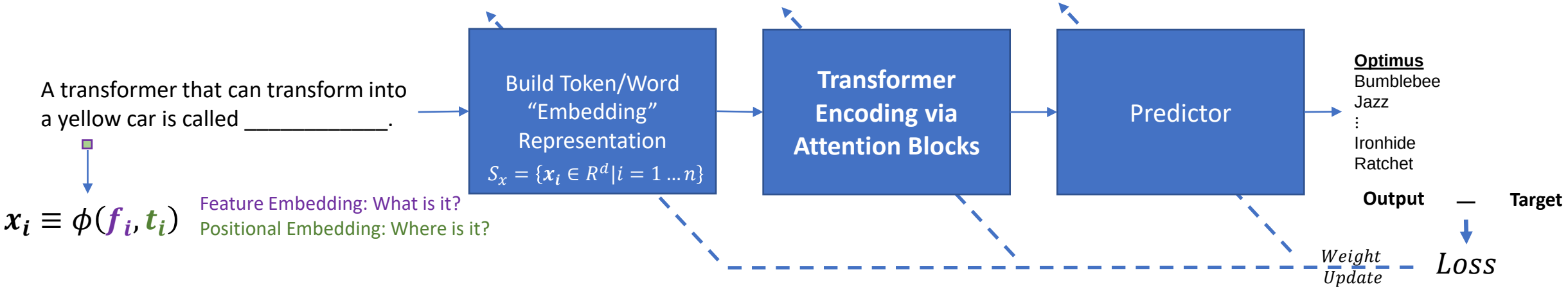
The dominant sequence transduction models are based on complex recurrent or convolutional neural networks in an encoder-decoder configuration. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

$$x'_q = A\left(x_q; M\left(x_q, S_{x_k}\right); \boldsymbol{\theta}\right) = \sum_{x_k \in M\left(x_q, S_{x_k}\right)} \mathrm{a}\left(x_q, x_k\,; M\left(x_q, S_{x_k}\right), \boldsymbol{\theta_k}\right) v(x_k; \boldsymbol{\theta_v}) = \sum_{x_k \in M\left(x_q, S_{x_k}\right)} \frac{\mathrm{k}\left(x_q, x_k; \boldsymbol{\theta_k}\right)}{\sum_{x_{k'} \in M\left(x_q, S_{x_k}\right)} \mathrm{k}\left(x_q, x_{k'}; \boldsymbol{\theta_k}\right)} v(x_k; \boldsymbol{\theta_v})$$

Tsai, Yao-Hung Hubert, Shaojie Bai, Makoto Yamada, Louis-Philippe Morency, and Ruslan Salakhutdinov. "**Transformer Dissection: A Unified Understanding of Transformer's Attention via the Lens of Kernel**." *ArXiv:1908.11775 [Cs, Stat]*, November 11, 2019. http://arxiv.org/abs/1908.11775.

Large Language Models: ChatGPT

- Simply: A transformer that generates the next word given some context
  - Multiple (>100) transformer layers with over a billion weights
  - Trained over the entire internet corpus
  - Fine-tuned (and controlled) with human feedback prompting

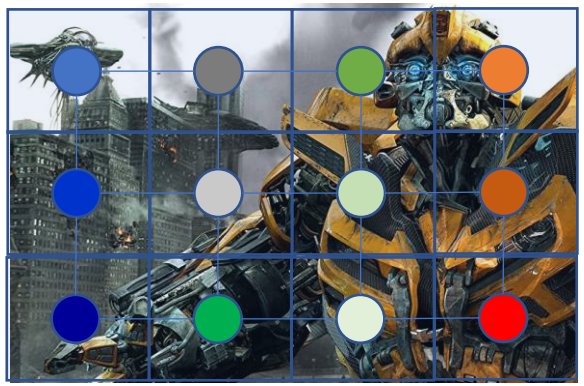


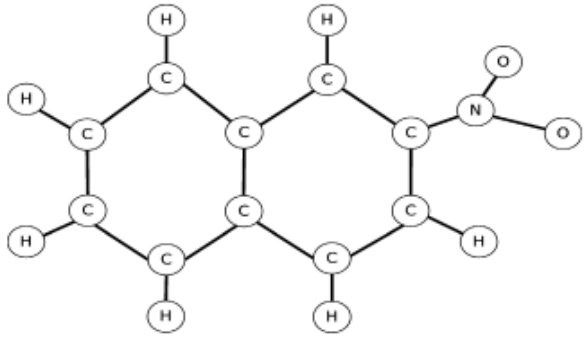
(NLP) Transformers (for next word prediction)

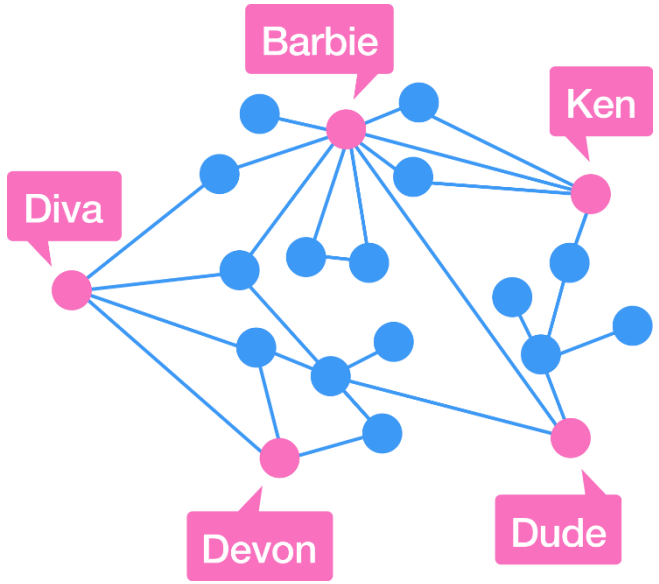
Simplest: $\phi(f_i, t_i) = f_i + t_i$

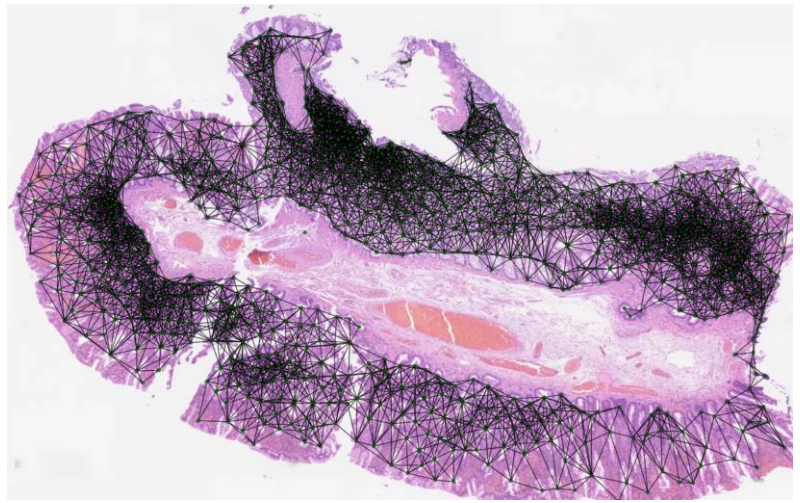
Graphs and their neural networks

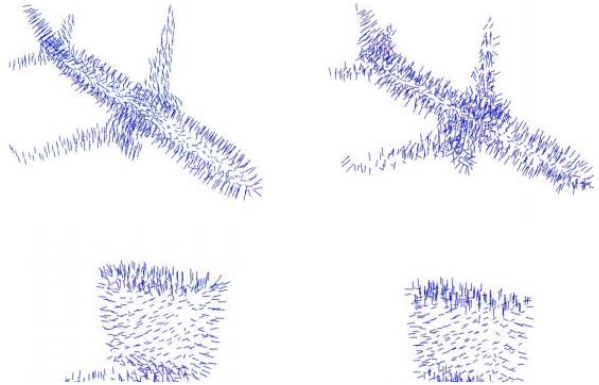
This is a graph





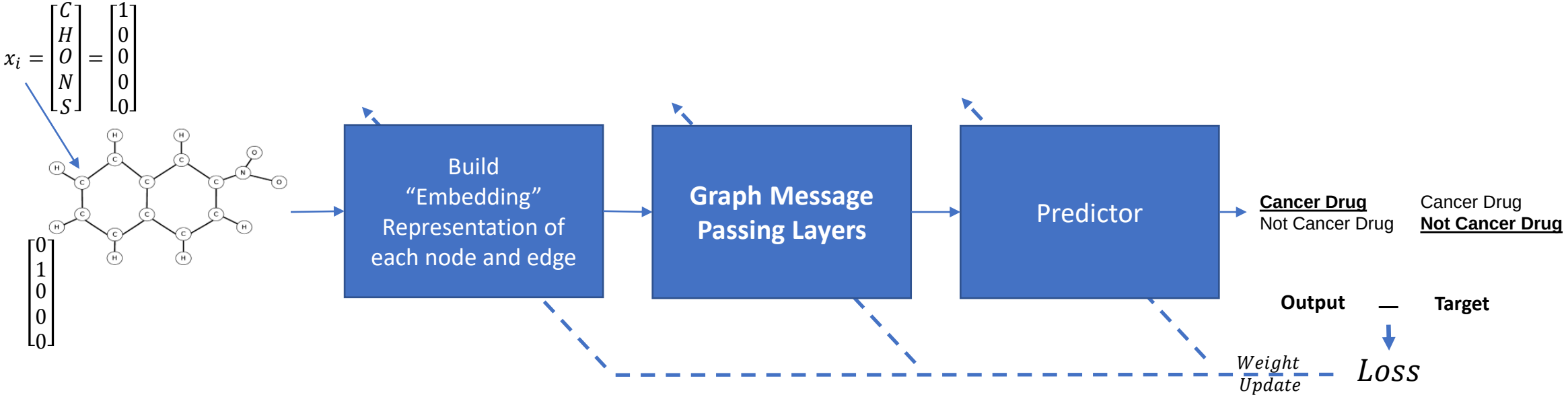






Graph Neural Networks

- Simple Graph Classification Example
  - Node and edge level prediction problems also possible



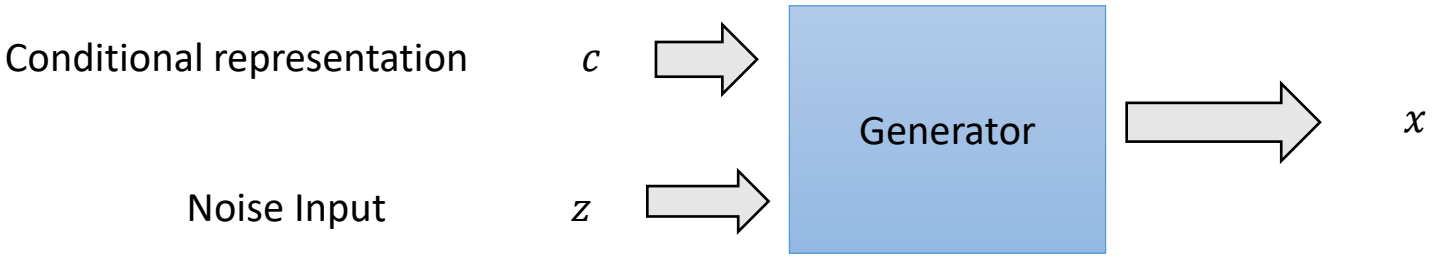
Input: Graph consisting of

Node set: what are things (each node has feature representation)

Edge set: how are they connected (each edge can have a feature representation but, in the very least, it tells us what nodes are connected by an edge)

Generative Machine Learning

“Image showing "It's coming home" in the context of the English team winning the football final”



https://github.com/foxtrotmike/CS909/blob/master/simpleGAN.ipynb
https://github.com/wgrgwrght/Simple-Diffusion/blob/main/SimpleDiffusion.ipynb


ENGLAND
IT'S COMING
HOME
adidas

Philosophical basis	Algorithms
I. Entities have (explicit or implicit) representations	Learning representations
II. Semantic relatedness of entities is context dependent and thus their representations are contextual	Using Convolutions, Transformers or Graph Layers
III. Representation of any entity can allow us to reconstruct or “generate” it	Generative Machine Learning: GANs, Latent Diffusion Models
IV. It is possible to develop representations in an inductive manner (through empirical observations)	Learning Algorithm: Optimization of model parameters through gradient descent based on existing data Learning mechanisms: Self Supervised Learning, Next word prediction
V. Intelligence is the capacity to develop and utilize causal representations of entities, enabling an organism or system to act effectively and adaptively.	Deep Reinforcement Learning?
VI. Empirical observations may not be enough	**Causal Machine Learning**

Next Steps

- Data Mining and Machine Learning Lecture Series
  - Assumes knowledge of programming
  - Covers Python
  - Assumes no knowledge of ML
  - Starts from simple classification
  - Ends at Generative Adversarial Networks

https://youtu.be/CjHp5HGckQk

https://sites.google.com/view/fayyaz/courses/data-mining