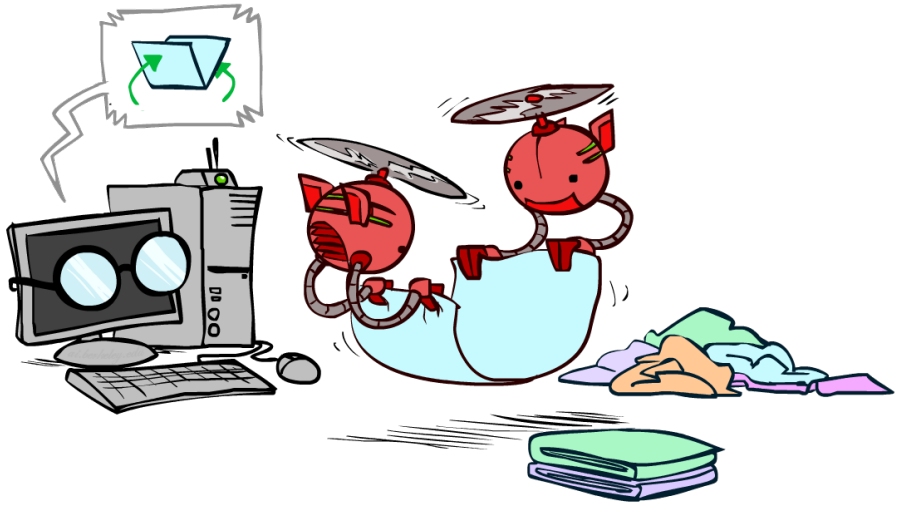
Deep Learning

CSE 473: Introduction to Artificial Intelligence



James Weichert (jpw@cs)
cs.uw.edu/473
acknowledgements

Welcome!
What's a deep thought you had recently?
sli.do **#cse473**

Administrivia

ANNOUNCEMENTS

- **Test 2** released

ASSIGNMENTS

- **Project 4 due** tomorrow (8.13)
- **Homework 4 due** next week (8.20)
  - No late period – HW4 must be turned in by 8.21

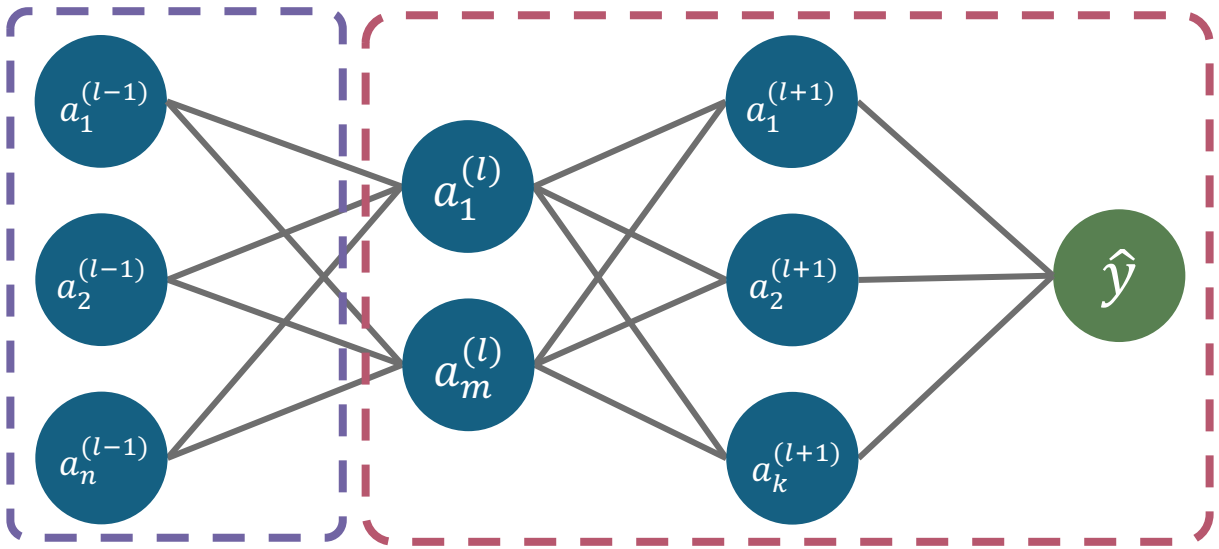
Neural Network Learning

DEFINITION

Compute the effect of a weight w on the loss function (i.e. $\frac{\partial \mathcal{L}}{\partial w}$) by **propagating the error backwards** through the network, combining with chain rule

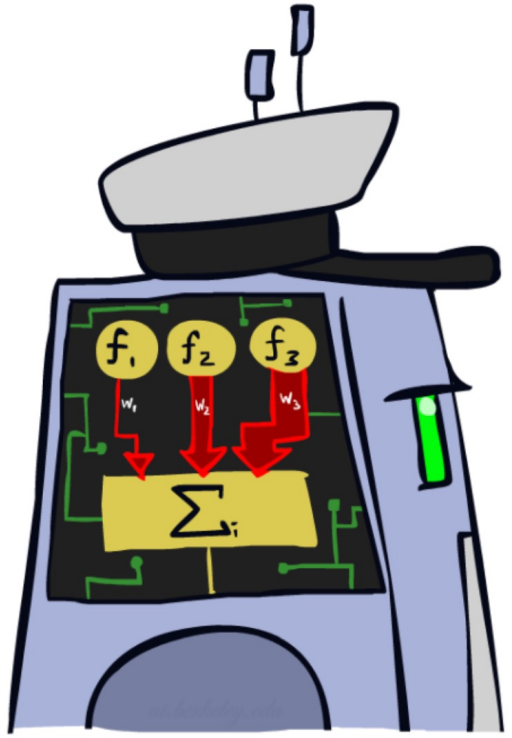
$$\frac{\partial}{\partial W^{(l)}} \mathcal{L}(\vec{x}, y) = \frac{\partial \mathcal{L}}{\partial \vec{z}^{(l)}} \cdot \frac{\partial \vec{z}^{(l)}}{\partial W^{(l)}} = \frac{\partial \mathcal{L}}{\partial \vec{z}^{(l)}} \cdot (\vec{a}^{(l-1)})^{\mathsf{T}}$$

$$\frac{\partial}{\partial A} A\vec{x} = \vec{x}^{\mathsf{T}}$$



Practical Considerations

FOUNDATIONS



- Derivative Calculations
  - Need to be able to calculate $\frac{\partial}{\partial z}\varphi(z) = \varphi'(z)$
  - What happens when neuron activations have very small gradients?
- Parallelism?
  - Matrix multiplication is a good candidate for parallelization
  - Deep learning breakthrough arrived after using GPUs for training
- Batching
  - Batch Gradient Descent: Standard gradient descent with all training samples
  - Stochastic Gradient Descent: Pick random training sample to backpropagate
- Scale Requirements?

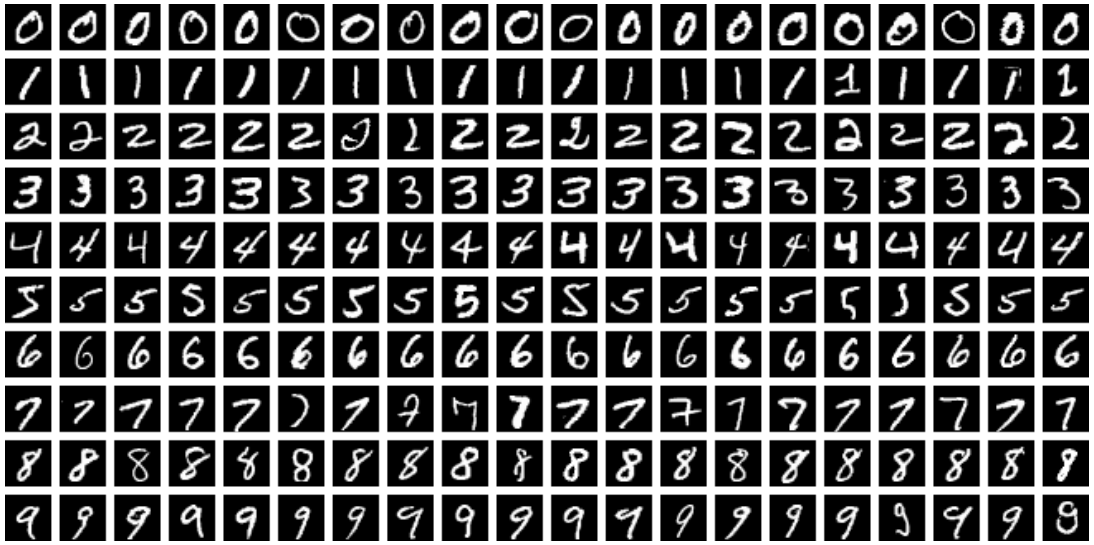
MLP Performance

CONTEXT

How do these models compare?

Model	Test Acc.	Train Time	Test Time
Logistic Reg.	92.55%	176.6s	0.01s
SVM (RBF)	96.85%	2.9s	7.74s
Decision Tree	87.58%	8.2s	0.01s
kNN (k=5)	96.88%	0.1s	3.58s
MLP	**97.92%**	45.9s	0.06s

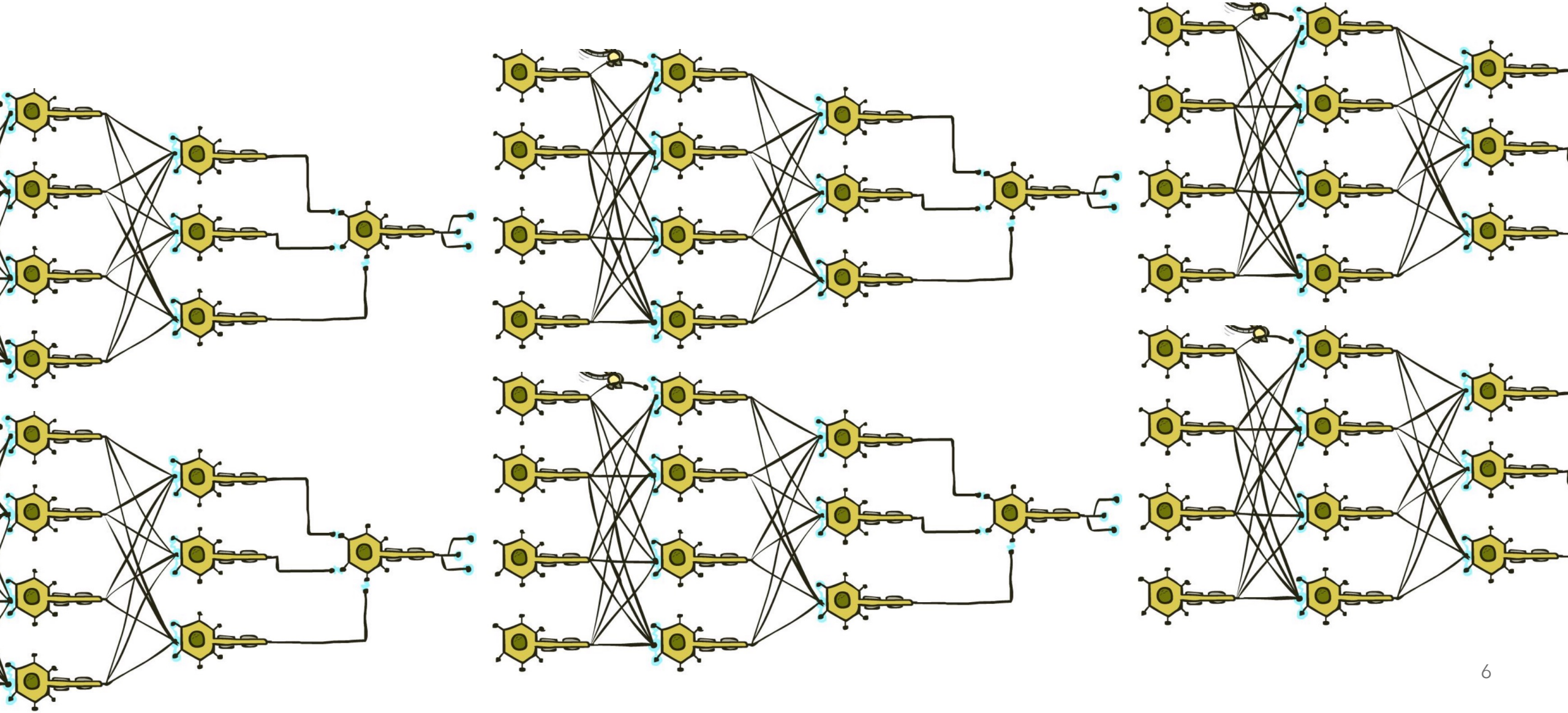
One-shot benchmark using standard sklearn implementations (no hyperparameter tuning). Python notebook generated by Claude Code.



Benchmark experiment Inspired by Michael Nielsen

Deep Networks

CONTEXT



Deep Network Training in Practice

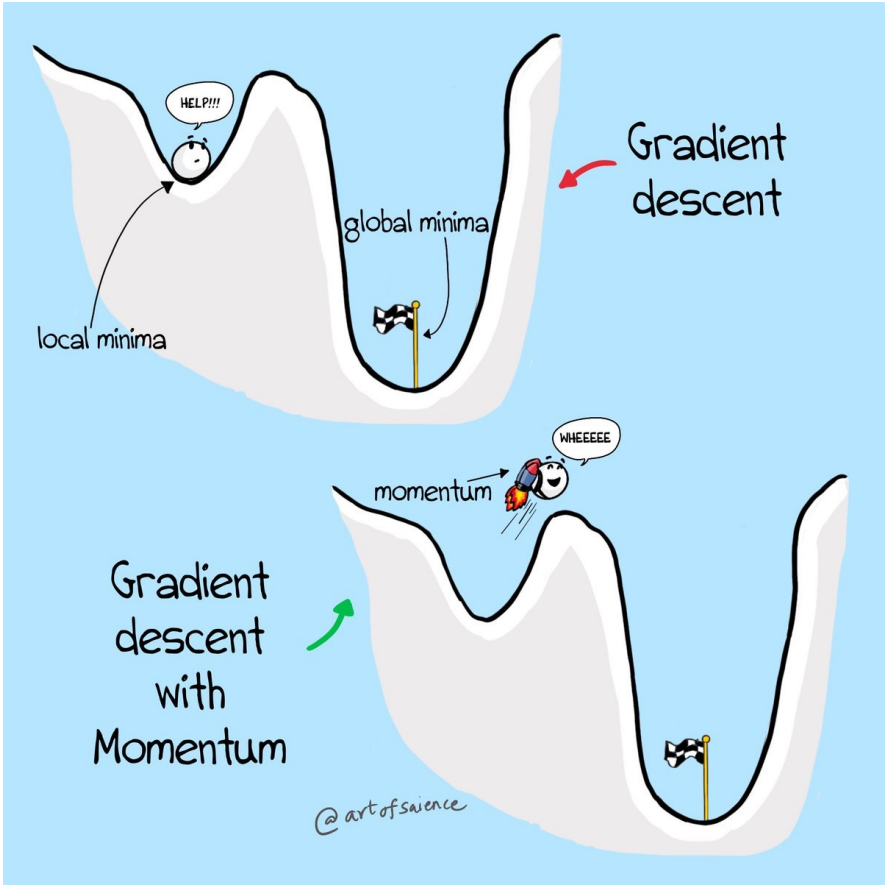
FOUNDATIONS

- Learning Rate η
  - Too large, overshoot minimum
  - Too small, slow to converge
- Momentum
  - Aims to overcome local minima and reduce oscillation
  - Adjust weight update:

$$v_t = \gamma \cdot v_{t-1} + \eta \frac{\partial \mathcal{L}}{\partial w}$$

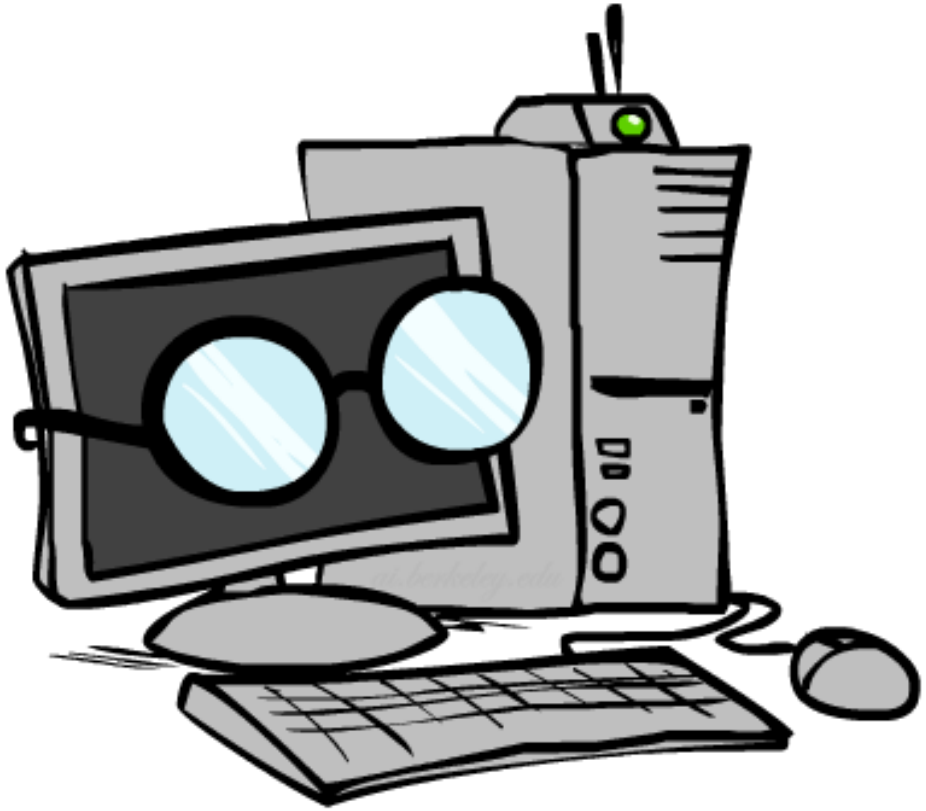
$$w \leftarrow w - v_t$$

- Vanishing Gradients

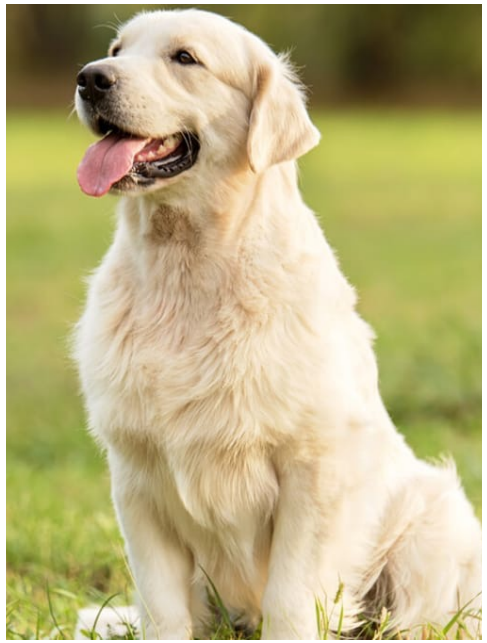


Questions?

live and on sli.do #cse473



Computer Vision

Cat or Dog?

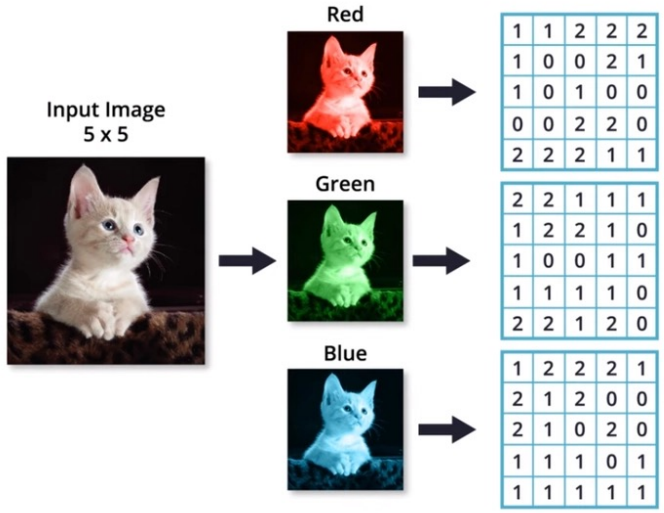
Images

CONTEXT

Images are *extremely* **high-dimensional** data

- Each pixel represents a part of the input
- For greyscale images, pixels encode brightness values for each x, y position
- For color (RGB) images, there are three input values per pixel representing each color **channel**

A 4032 x 3024 pixel smartphone photo has 36.5m input values



Goal: Leverage structure of images to reduce complexity

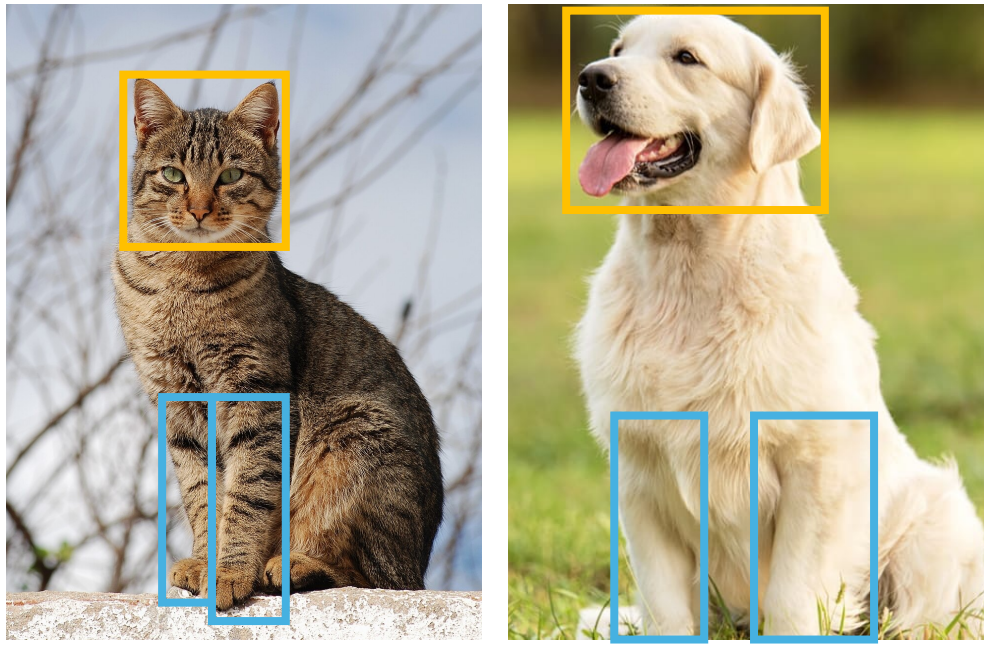
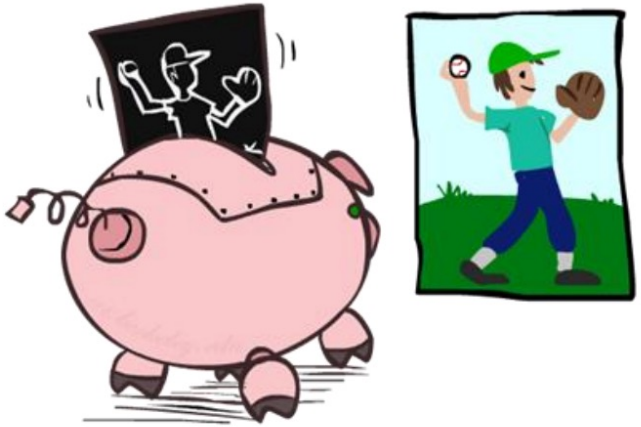


Image Features

Manual Feature Engineering Is Hard…



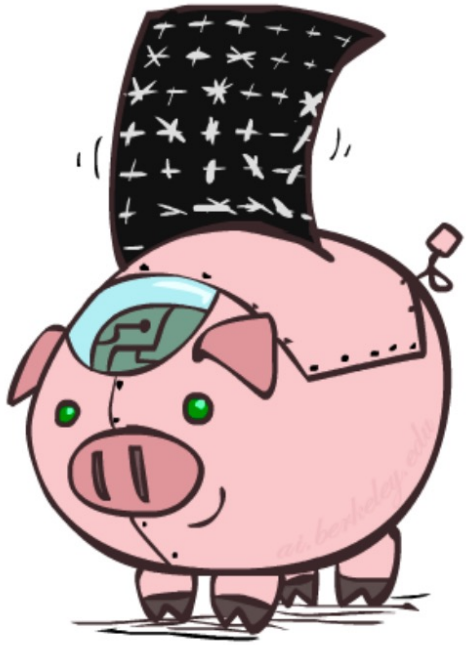
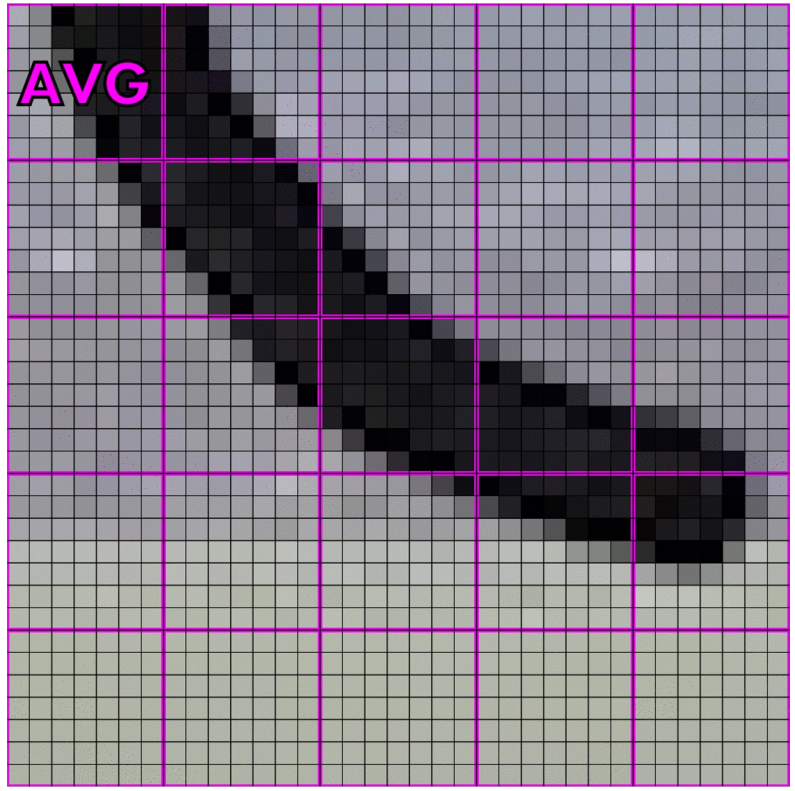


Image Transformation

FOUNDATIONS

Downsampling

EXAMPLE

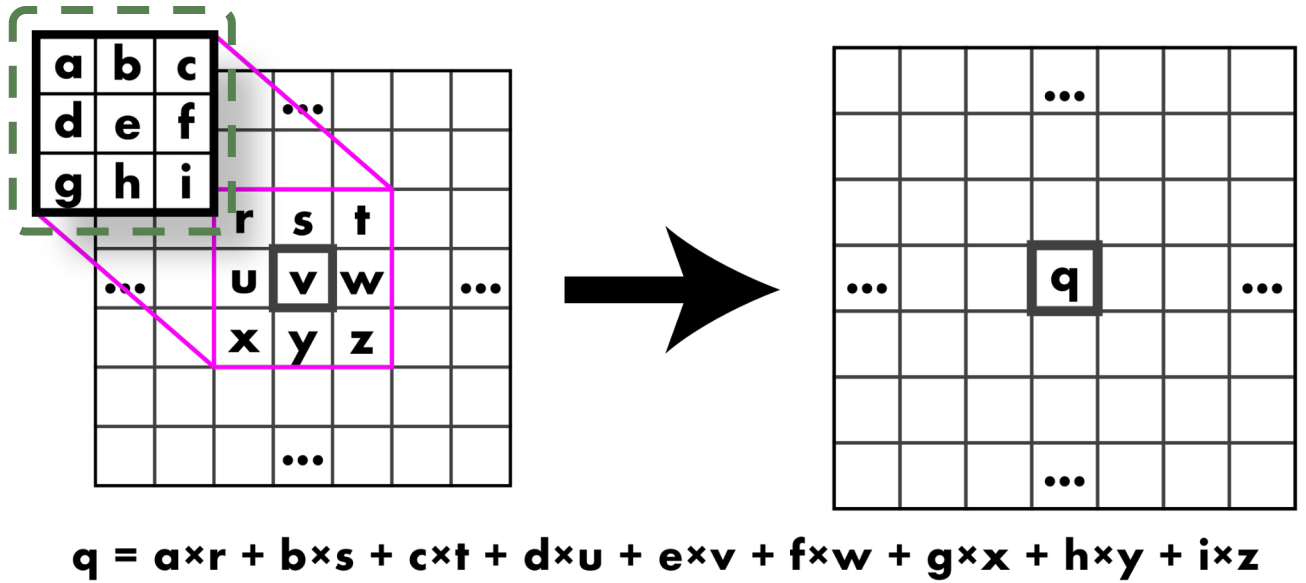


Convolution

DEFINITION

In computer vision, a **convolution** is an image transformation operation that involves sliding a small **kernel** (filter) over the image

$$I * K(i,j) = \sum_m \sum_n I(i+m, j+n) K(m,n)$$



QUESTIONS

Questions?

live and on sli.do #cse473

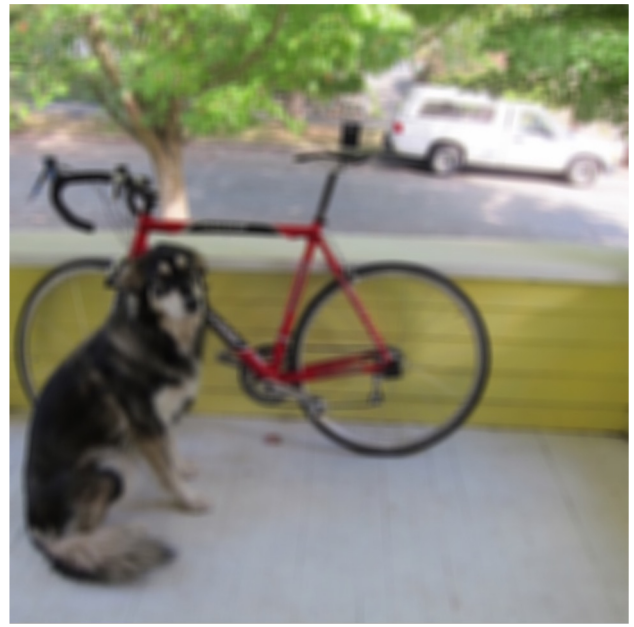
Blurry Image

EXAMPLE

$\frac{1}{49}$

1×	1×	1×	1×	1×	1×	1×
1×	1×	1×	1×	1×	1×	1×
1×	1×	1×	1×	1×	1×	1×
1×	1×	1×	1×	1×	1×	1×
1×	1×	1×	1×	1×	1×	1×
1×	1×	1×	1×	1×	1×	1×
1×	1×	1×	1×	1×	1×	1×

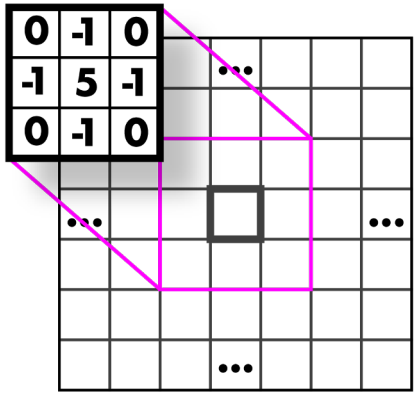
$*$



Mystery Kernel

EXAMPLE

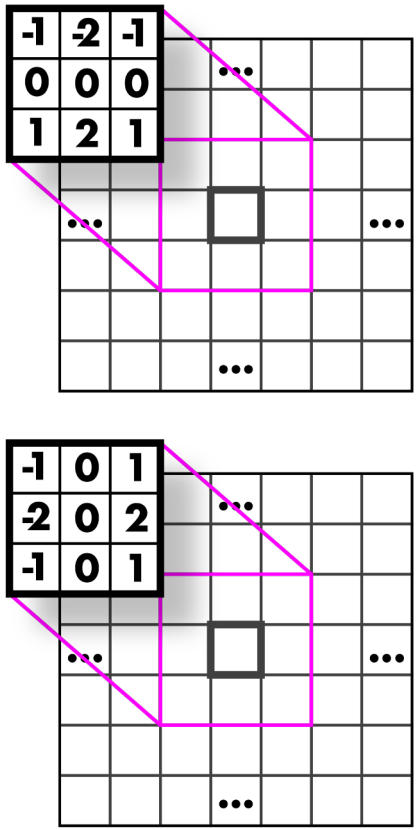
Amplify current pixel, add opposite of adjacent pixels

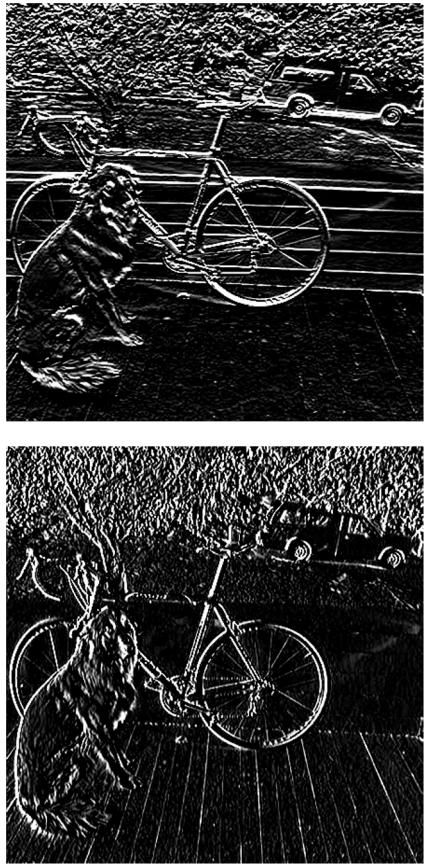


Sharpen

Abstracting Images

FOUNDATIONS

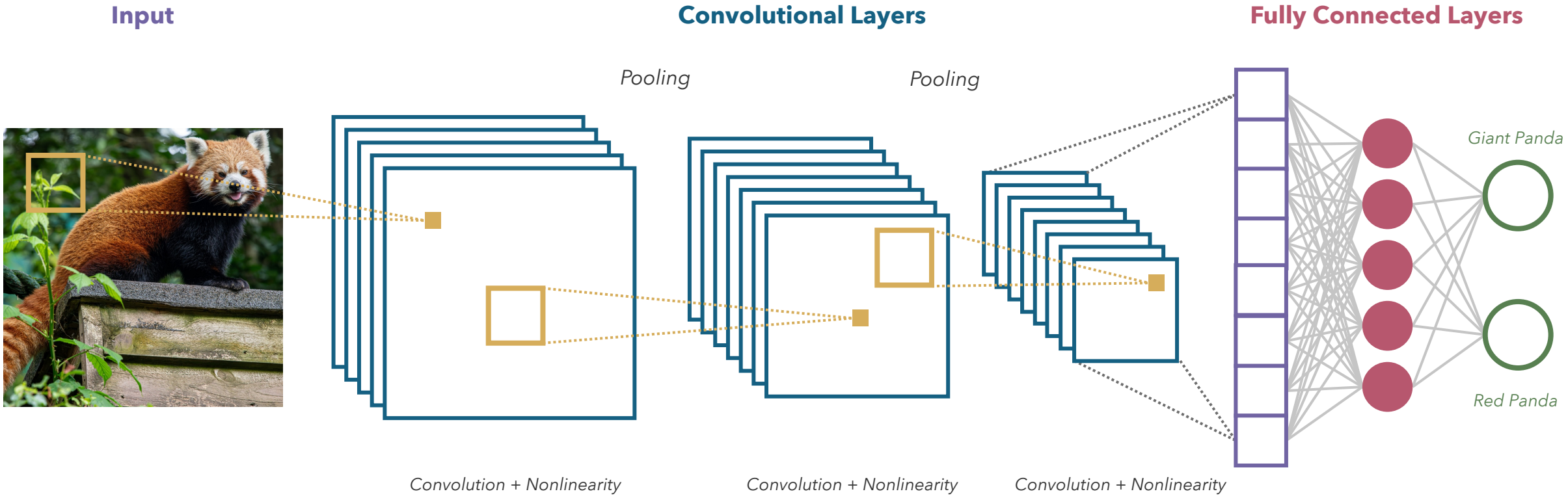




Sobel Kernels

Convolutional Neural Networks

FOUNDATIONS



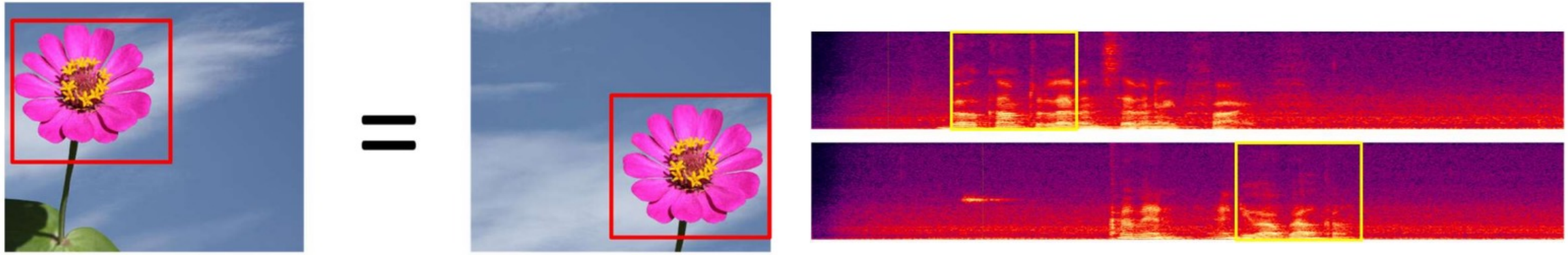
Pooling

DEFINITION

Pooling downsizes the input by aggregating values in each region

Shift Invariance

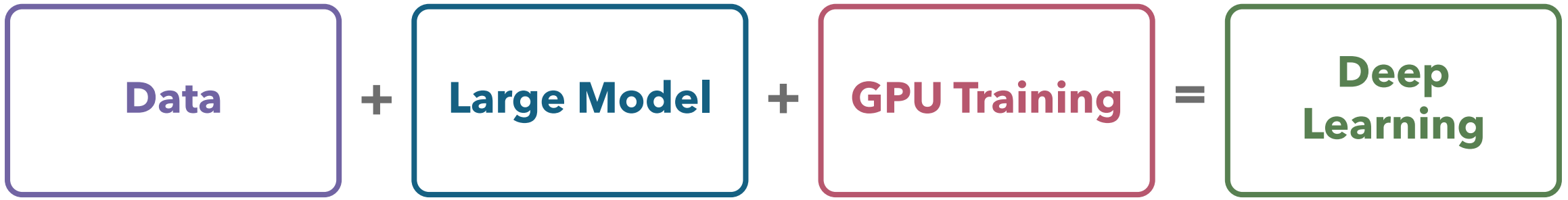
FOUNDATIONS



AlexNet

CONTEXT

"ImageNet Classification with Deep Convolutional Neural Networks" by Krizhevsky, Sutskever, and Hinton (2012) kickstarted a deep learning revolution



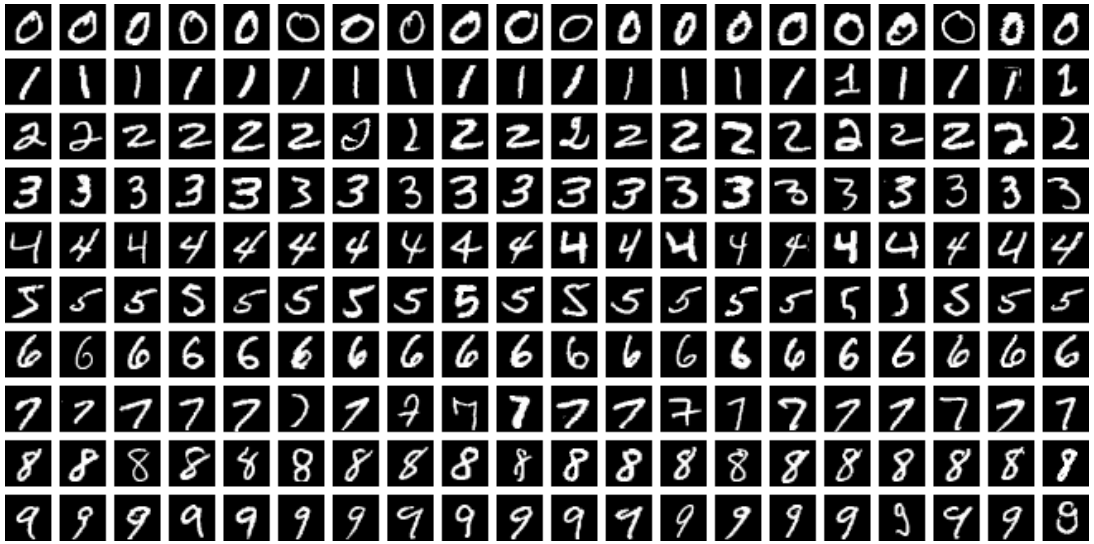
CNN Performance

CONTEXT

How do these models compare?

Model	Test Acc.	Train Time	Test Time
Logistic Reg.	92.55%	176.6s	0.01s
SVM (RBF)	96.85%	2.9s	7.74s
Decision Tree	87.58%	8.2s	0.01s
kNN (k=5)	96.88%	0.1s	3.58s
MLP	97.92%	45.9s	0.06s
CNN	**98.95%**	31.0s	0.42s

One-shot CNN benchmark using PyTorch implementation (no hyperparameter tuning). Python notebook generated by Claude Code.

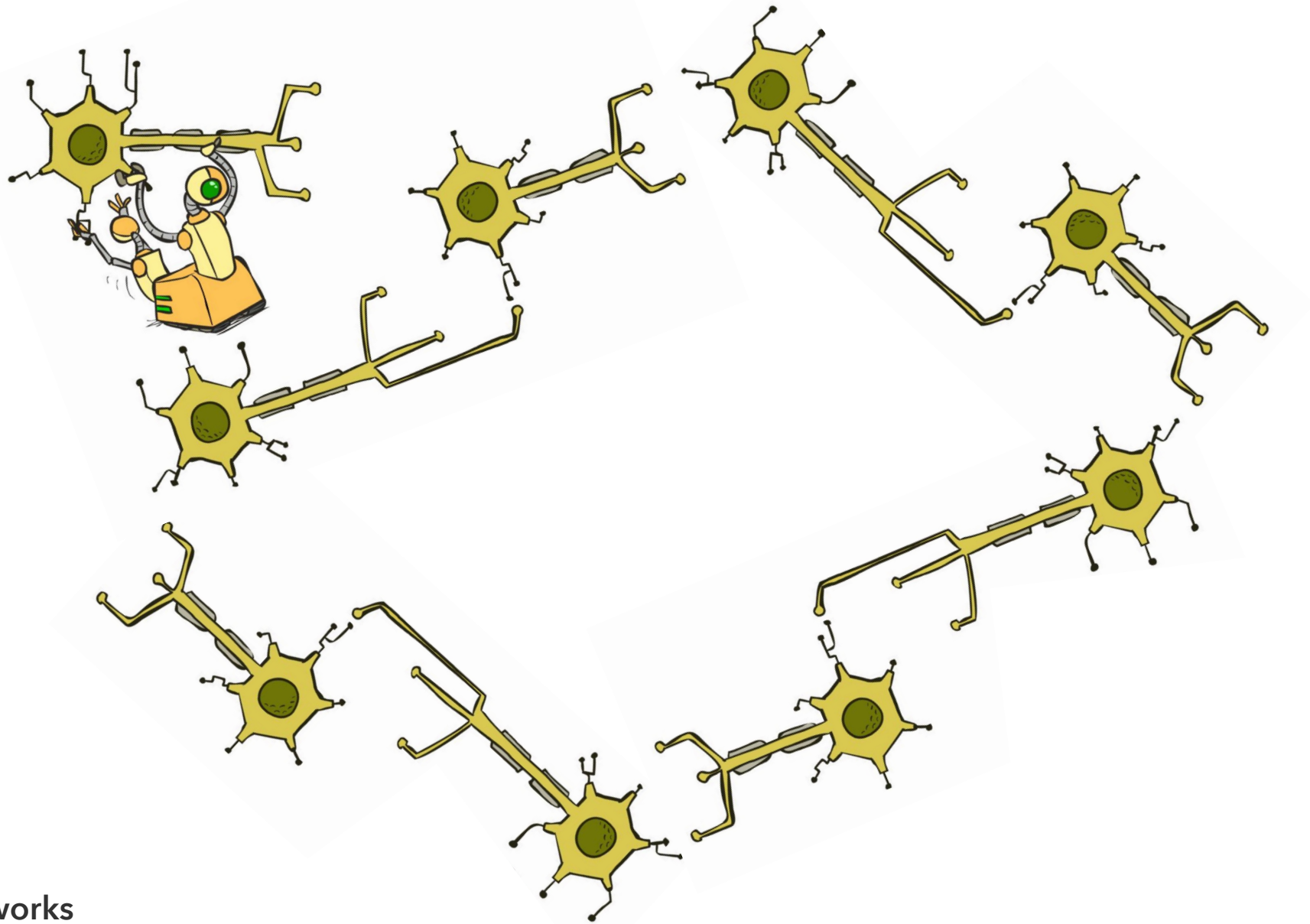


Benchmark experiment Inspired by Michael Nielsen

QUESTIONS

Questions?

live and on sli.do #cse473



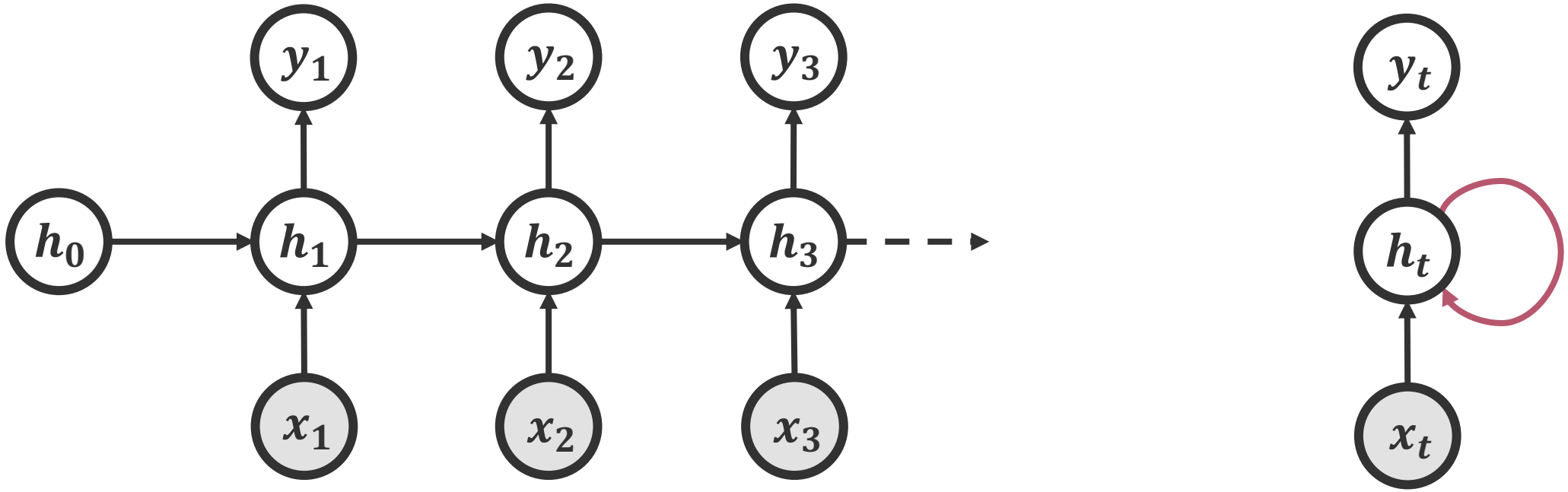
Recurrent Networks

Dependent Sequences

CONTEXT

Prediction at time t depends on prediction at time $t-1$, etc.

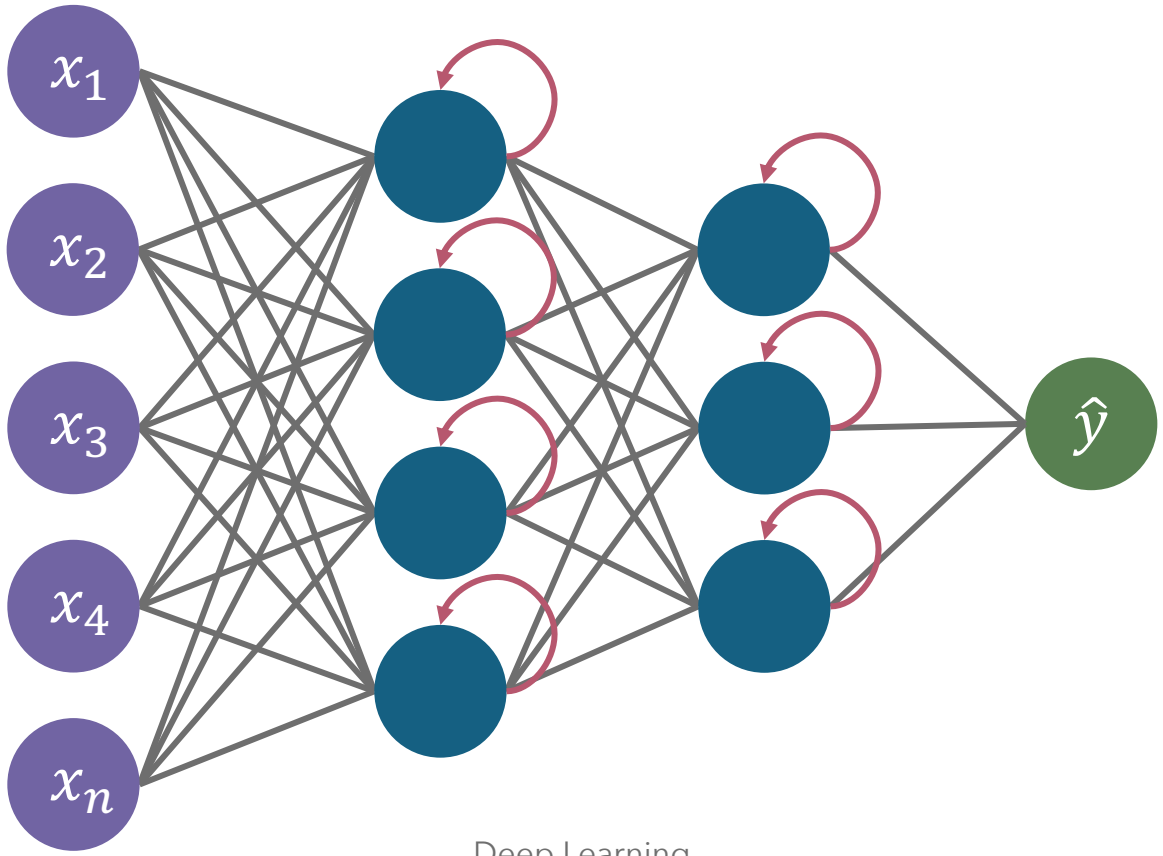
- Time series data, moving images, *language*



Recurrent Neural Networks

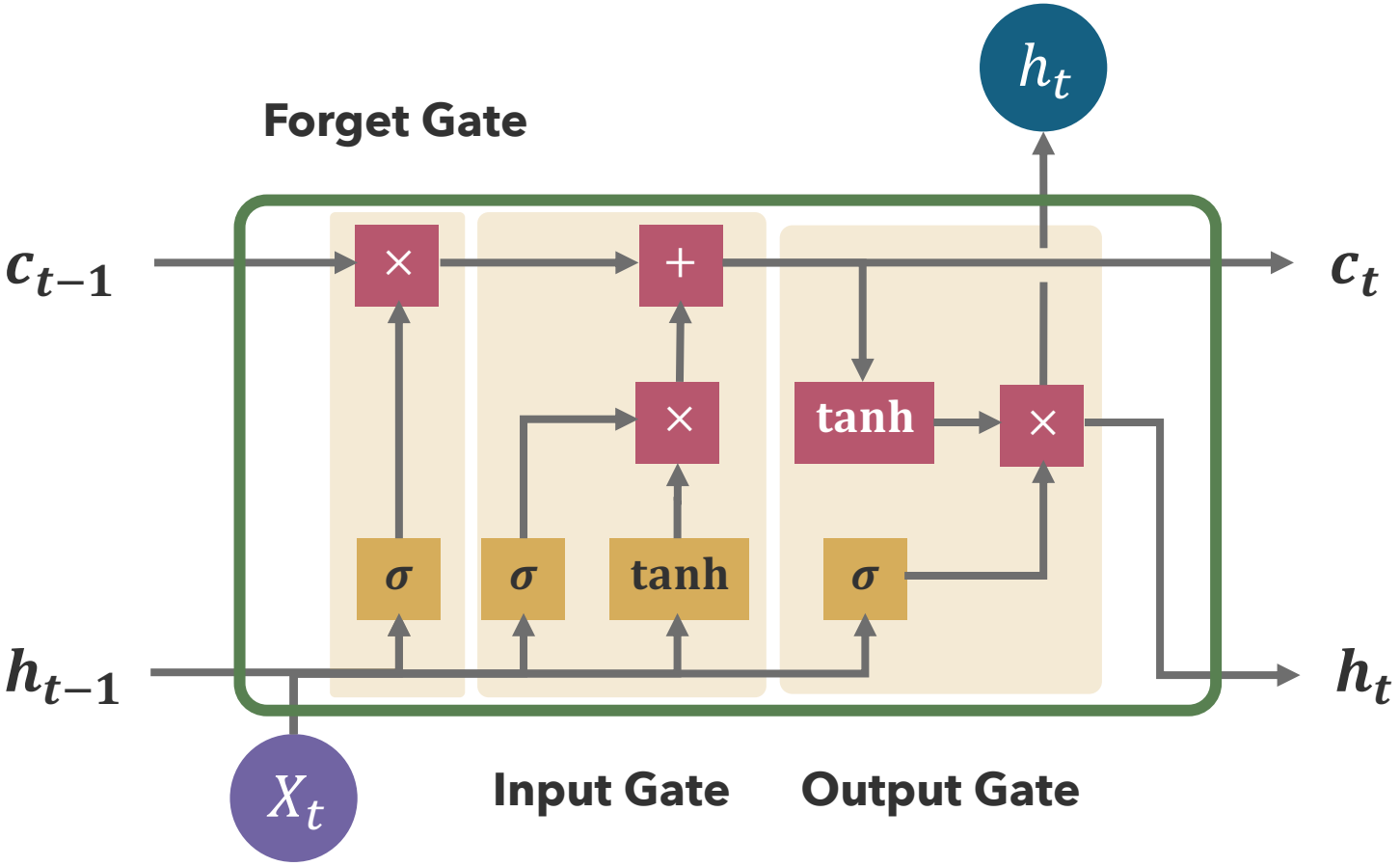
FOUNDATIONS

Recurrent Neural Networks include backwards (recurrent) connections



Long Short Term Memory (LSTM)

DEFINITION



Questions?

live and on sli.do #cse473

that's it for today!

SUMMARY

- CNNs learn convolutional kernels to extract local features from images to use in fully connected layers
- Recurrent networks use previous outputs to influence subsequent predictions
- Language is complex, requiring flexible 'focus'

UPCOMING

- Natural Language Processing
- Large Language Models

REMINDERS

- Complete **Practice Problem 19**
- Do **Project 4** (due 8.13)
- Do **Homework 4** (due 8.20)