# Introduction to Deep Learning

Nandita Bhaskhar

(content adapted from past CS229 teams)

Nov 5th, 2021

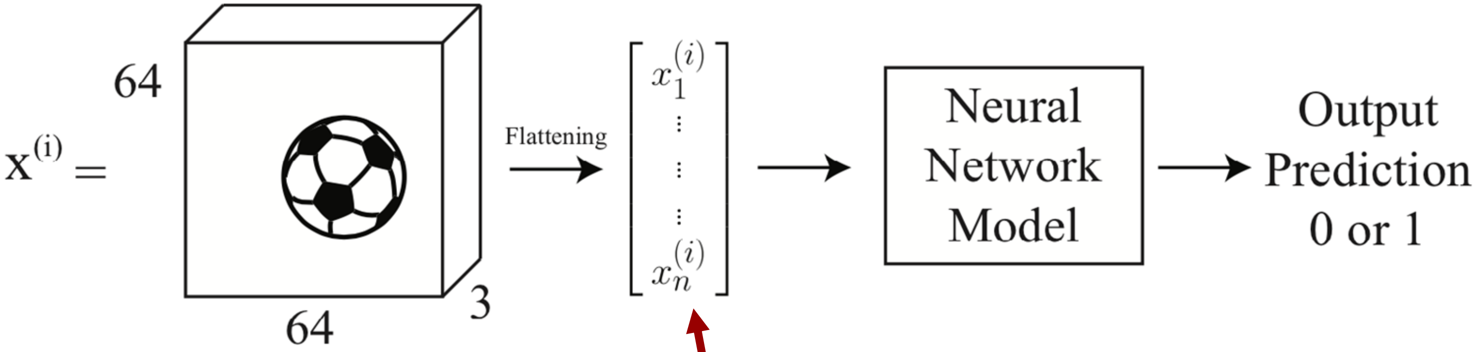
# Overview

- Motivation for deep learning
- Areas of Deep Learning
- Convolutional neural networks
- Recurrent neural networks
- Deep learning tools

# But we learned multi-layer perceptron in class?

Expensive to learn.  Will not generalize well

Does not exploit the order and local relations in the data!



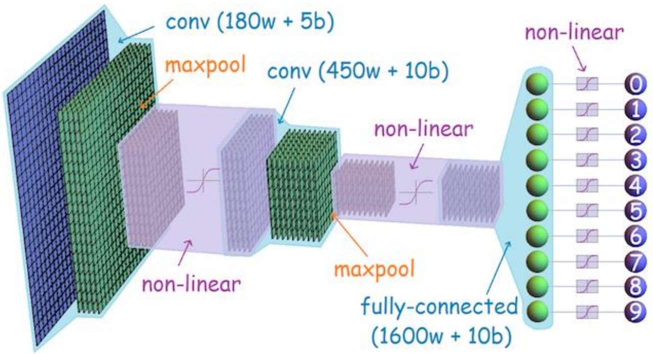

64x64x3=12288 parameters
We also want **many** layers

# Overview

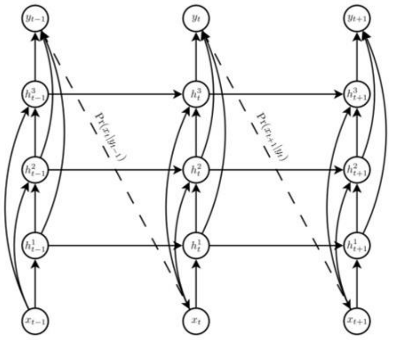
- Motivation for deep learning
- Areas of Deep Learning
- Convolutional neural networks
- Recurrent neural networks
- Deep learning tools

# What are areas of deep learning?

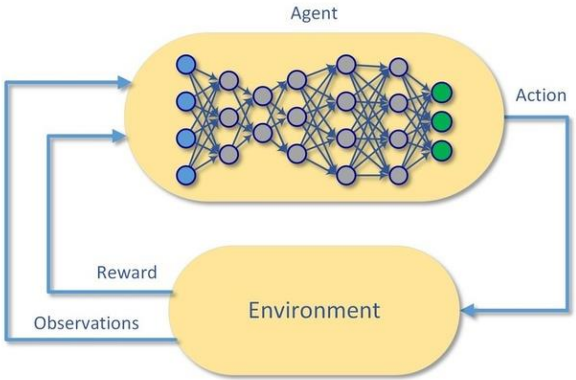
Convolutional NN
Image



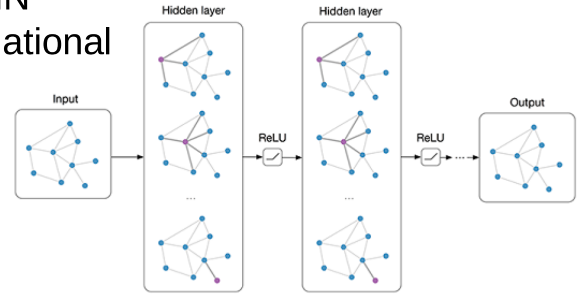

Recurrent NN
Time Series



Deep RL
Control System




Graph NN
Networks/Relational

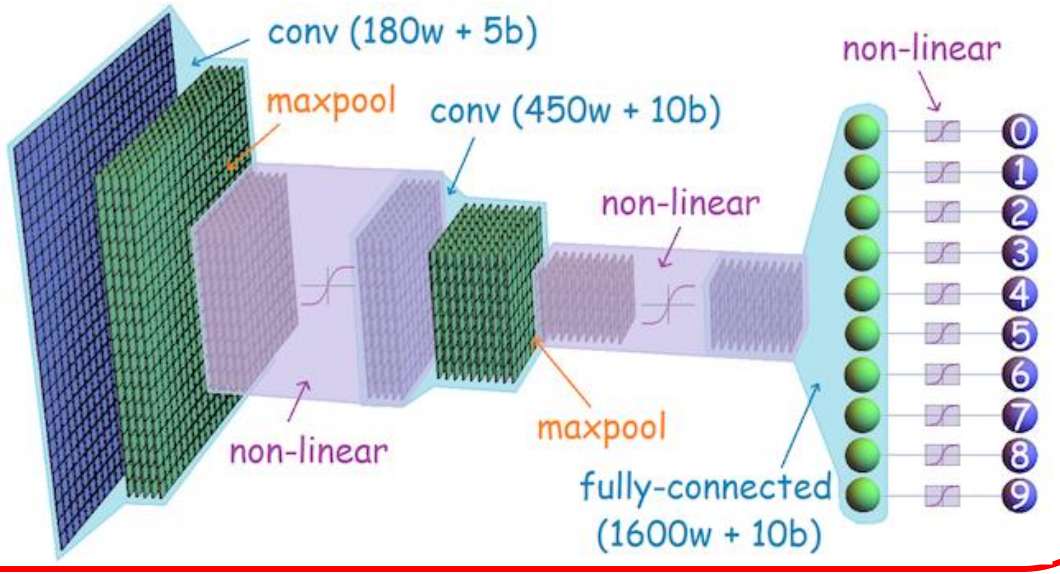


# Overview

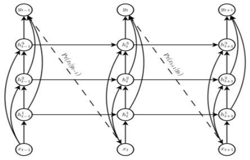
- Motivation for deep learning
- Areas of Deep Learning
- Convolutional neural networks
- Recurrent neural networks
- Deep learning tools

# What are areas of deep learning?

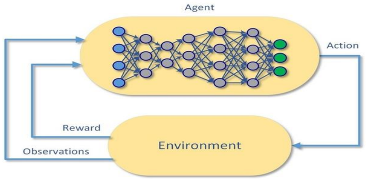
Convolutional Neural Network



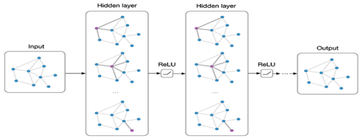

Recurrent NN



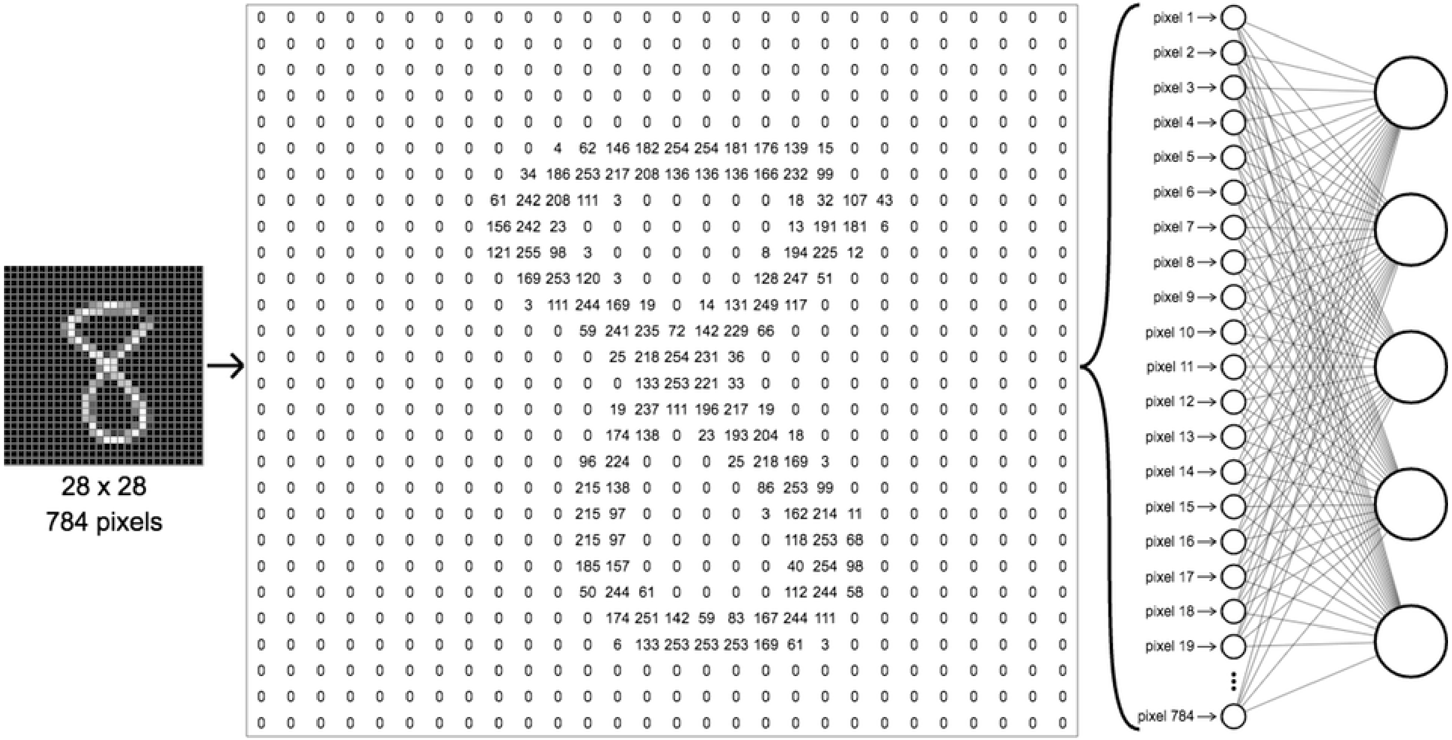
Deep RL




Graph NN

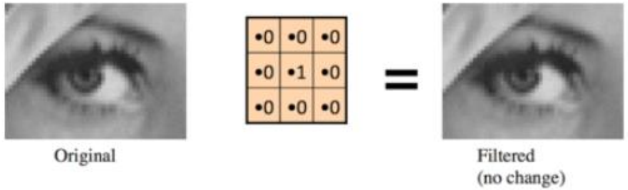


# Let us look at images in detail

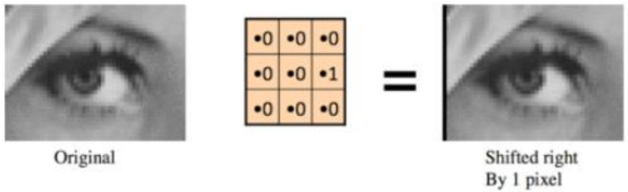


# Filters

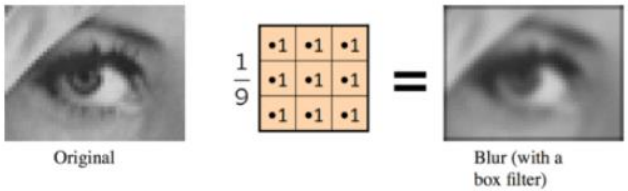
No change:




Shifted right by one pixel:



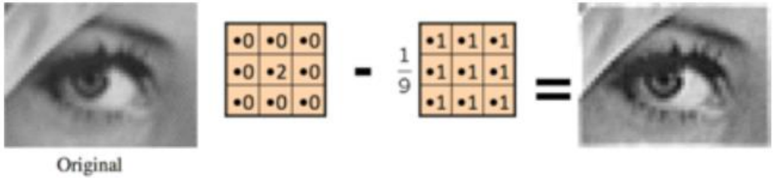

Blurred (you already saw this above):




Note the edge artifact.*

https://ai.stanford.edu/~syyeung/cvweb/tutorials.html

Sharpening




https://ai.stanford.edu/~syyeung/cvweb/tutorials.html

Edge Detection: Laplacian Filters

| | | |
|---|---|---|
| 0 | -1 | 0 |
| -1 | 4 | -1 |
| 0 | -1 | 0 |

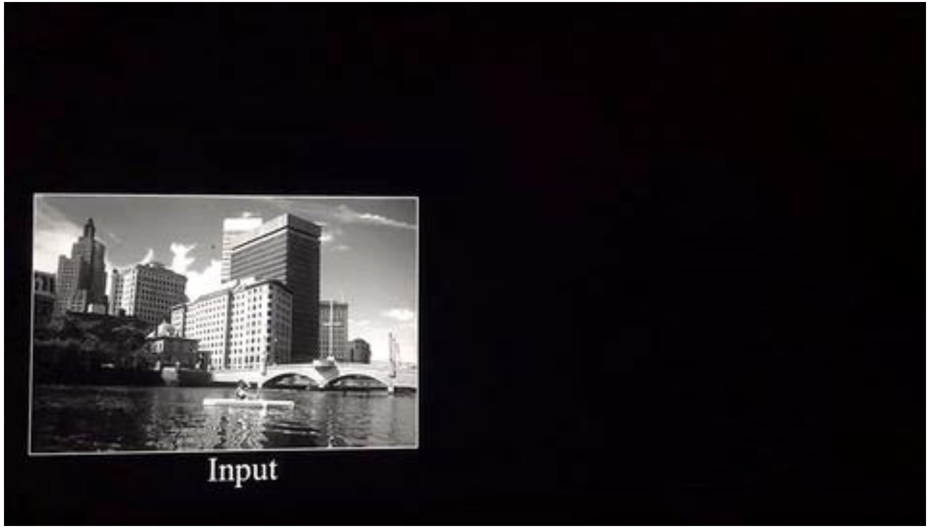
| | | |
|---|---|---|
| -1 | -1 | -1 |
| -1 | 8 | -1 |
| -1 | -1 | -1 |

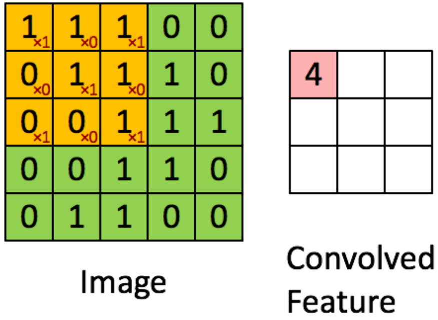
# Filters

Why not extract features using filters?

Better, why not let the data dictate what filters to use?

Learnable filters!!



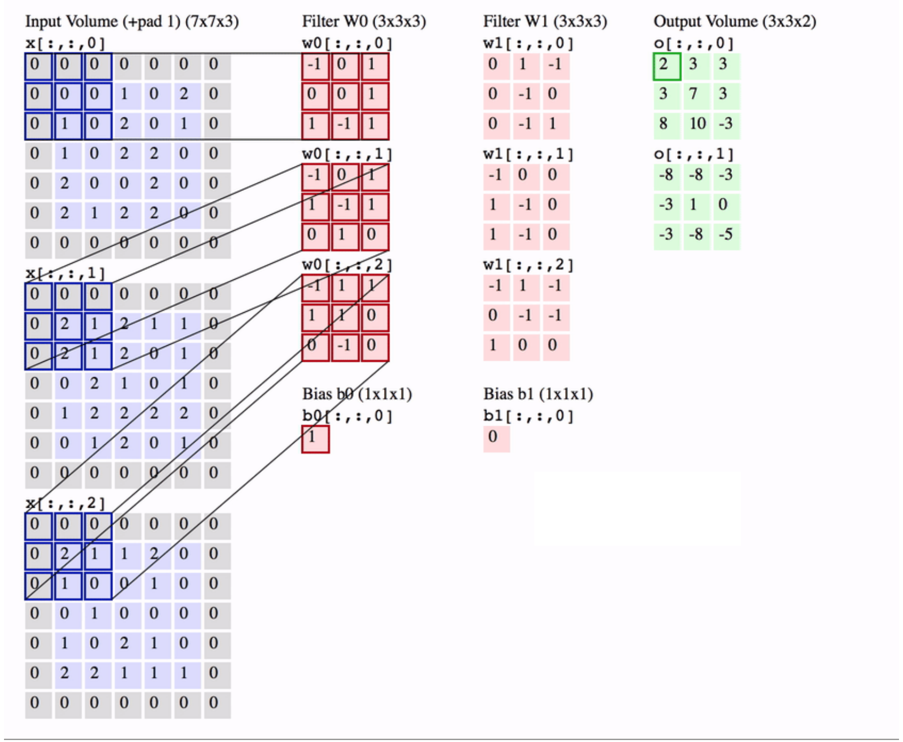
# Convolution on multiple channels

Images are generally RGB !!

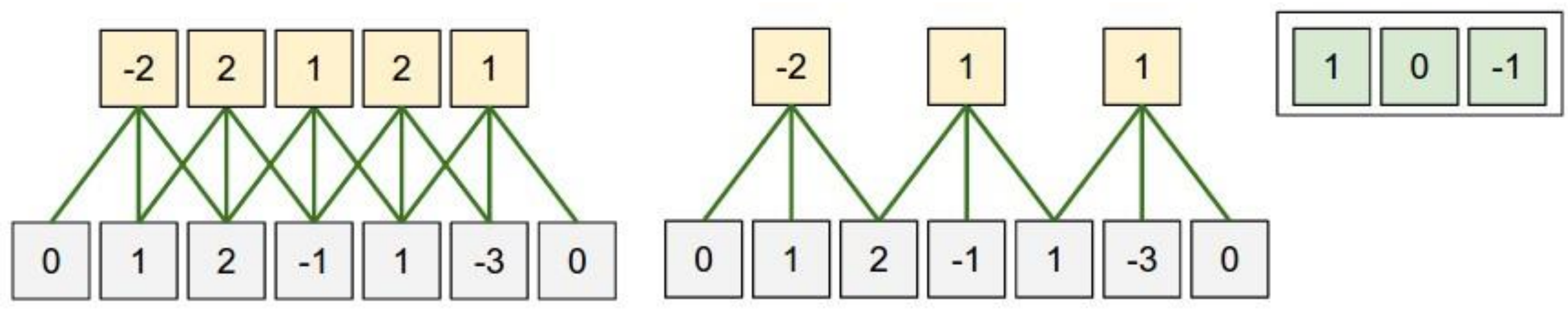
How would a filter work on a image with RGB channels?

The filter should also have 3 channels.

Now the output has a channel for every filter we have used.



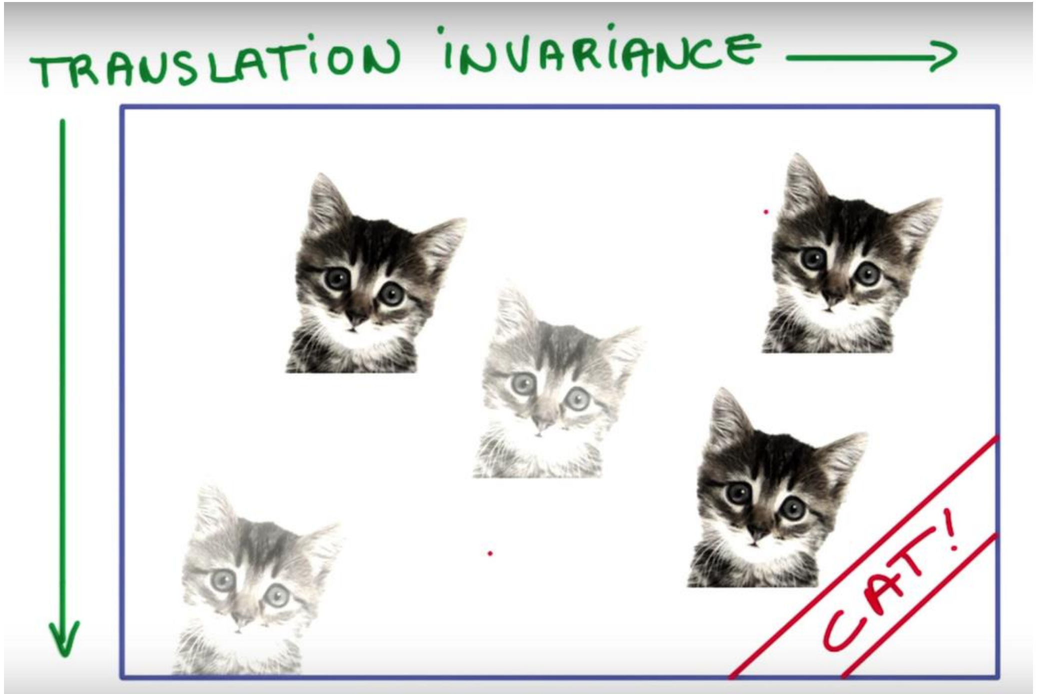
# Parameter Sharing



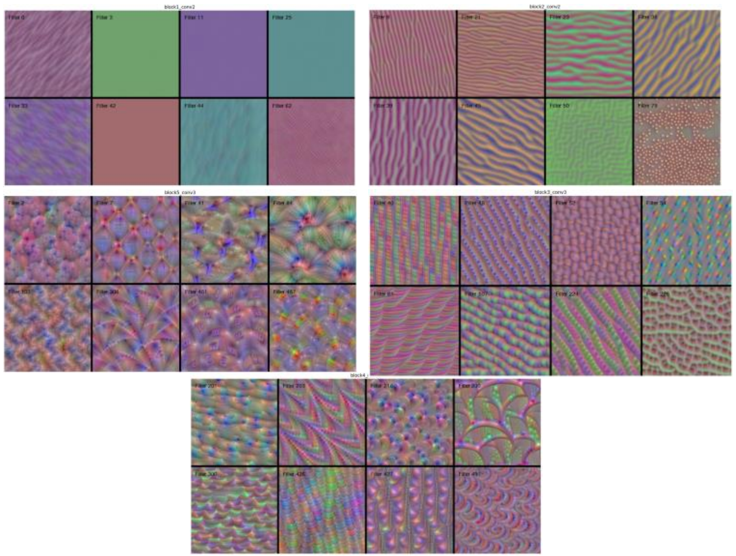

Lesser the parameters less computationally intensive the training. This is a win win as we are reusing parameters.

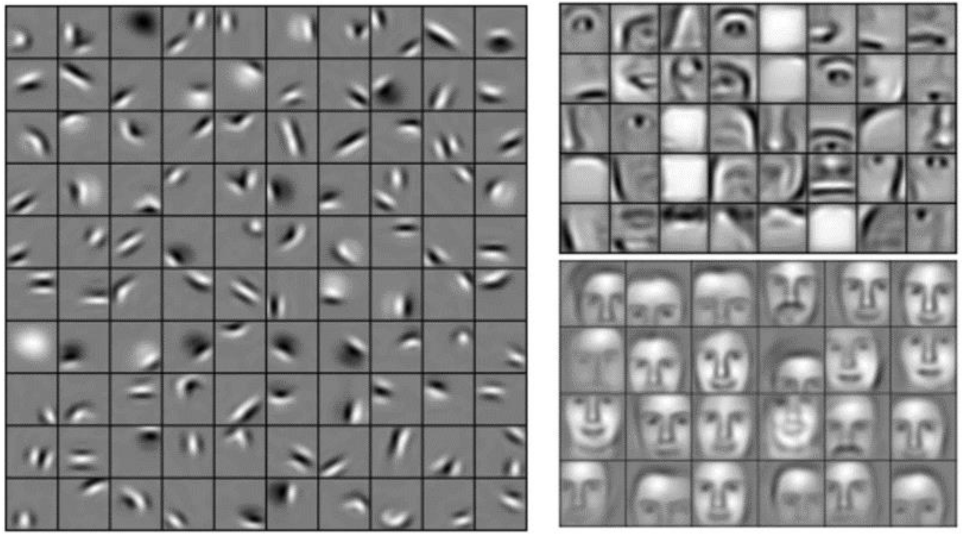
# Translational invariance

Since we are training filters to detect cats and the moving these filters over the data, a differently positioned cat will also get detected by the same set of filters.



# Filteres? Layers of filters?

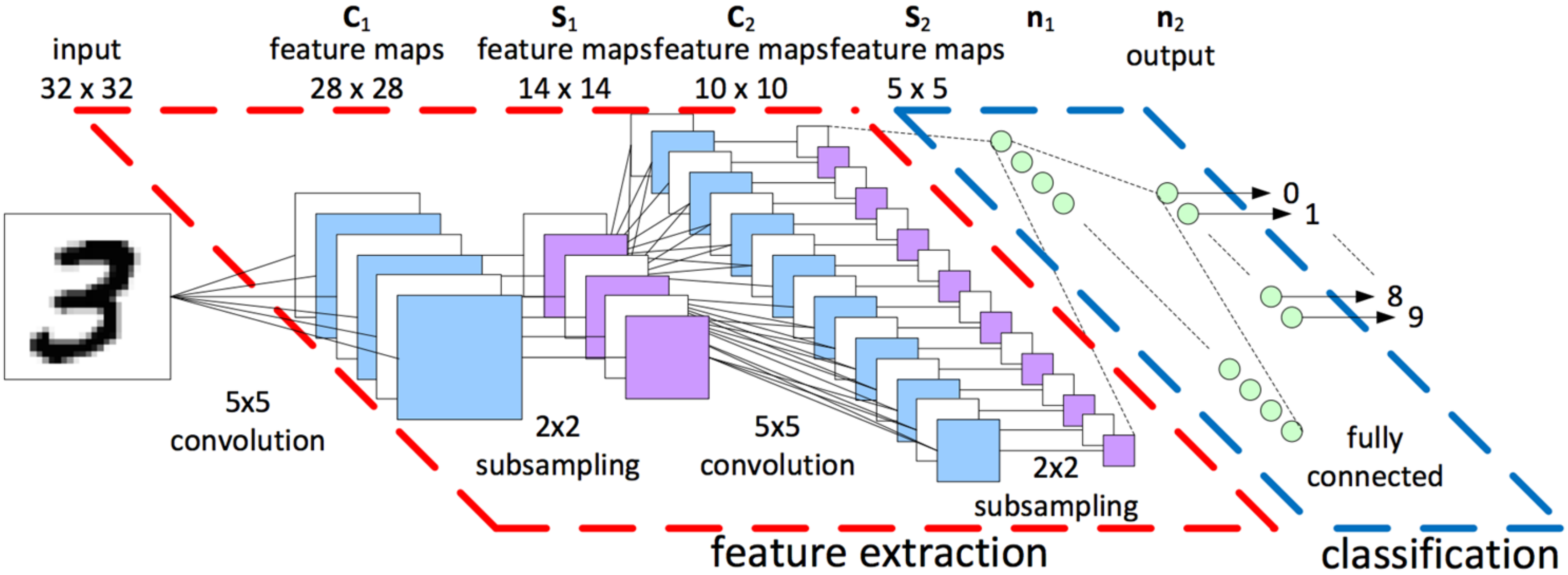




Images that maximize filter outputs at certain layers. We observe that the images get more complex as filters are situated deeper

How deeper layers can learn deeper embeddings. How an eye is made up of multiple curves and a face is made up of two eyes.

# How do we use convolutions?




Let convolutions extract features!

Image credit: LeCun et al. (1998)

# Convolution really is just a linear operation

In fact convolution is a giant matrix multiplication.

We can expand the 2 dimensional image into a vector and the conv operation into a matrix.

$$\begin{pmatrix} k1 & k2 & 0 & k3 & k4 & 0 & 0 & 0 & 0 \\ 0 & k1 & k2 & 0 & k3 & k4 & 0 & 0 & 0 \\ 0 & 0 & 0 & k1 & k2 & 0 & k3 & k4 & 0 \\ 0 & 0 & 0 & 0 & k1 & k2 & 0 & k3 & k4 \end{pmatrix} . \begin{pmatrix} x1 \\ x2 \\ x3 \\ x4 \\ x5 \\ x6 \\ x7 \\ x8 \\ x9 \end{pmatrix}$$

$$\begin{pmatrix} x1 & x2 & x3 \\ x4 & x5 & x6 \\ x7 & x8 & x9 \end{pmatrix} * \begin{pmatrix} k1 & k2 \\ k3 & k4 \end{pmatrix}$$

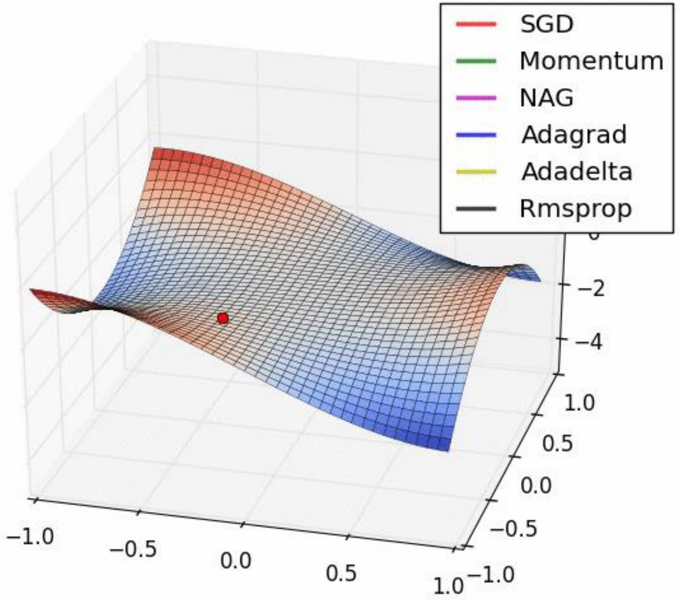
$$\begin{pmatrix} k1\,x1 + k2\,x2 + k3\,x4 + k4\,x5 \\ k1\,x2 + k2\,x3 + k3\,x5 + k4\,x6 \\ k1\,x4 + k2\,x5 + k3\,x7 + k4\,x8 \\ k1\,x5 + k2\,x6 + k3\,x8 + k4\,x9 \end{pmatrix}$$

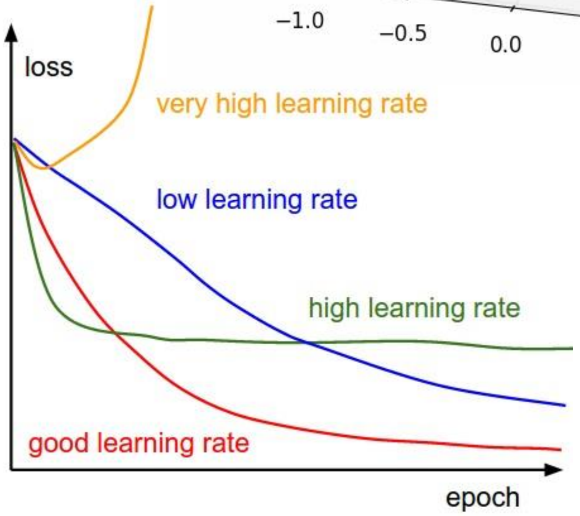
# How do we learn?

Instead of $\theta := \theta + \alpha \left(y^{(i)} - h_\theta(x^{(i)})\right) x^{(i)}$

They are “optimizers”

- Momentum: Gradient + Momentum
- Nestrov: Momentum + Gradients
- Adagrad: Normalize with sum of sq
- RMSprop: Normalize with moving avg of sum of squares
- ADAM: RMsprop + momentum



# Mini-batch Gradient Descent

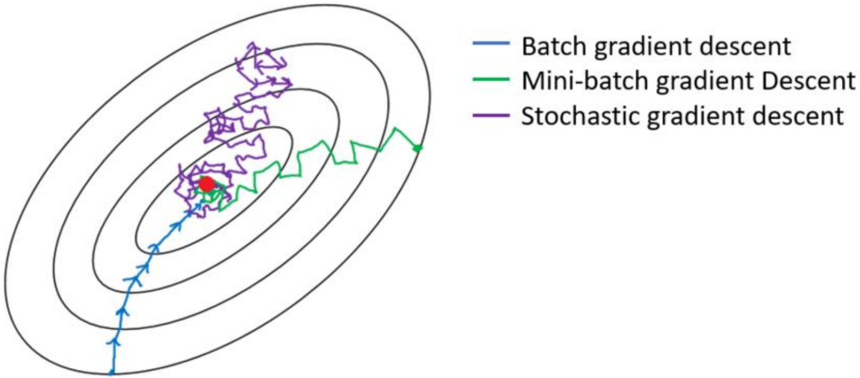
Expensive to compute gradient for large dataset

Memory size

Compute time

Mini-batch: takes a sample of training data

How to we sample intelligently?

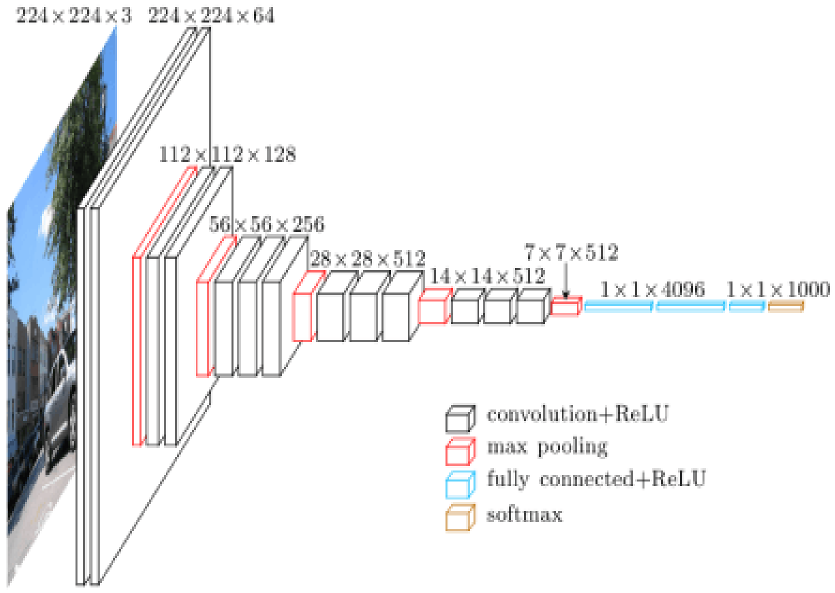


# Is deeper better?

Deeper networks seem to be more powerful but harder to train.

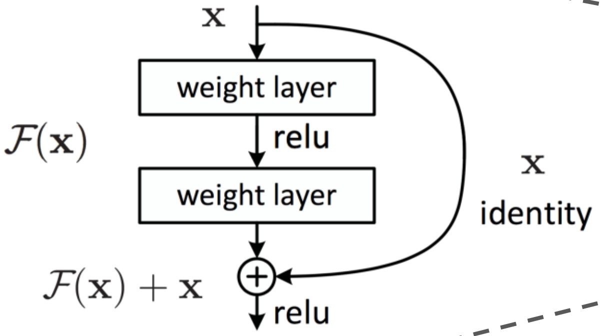
- Loss of information during forward propagation
- Loss of gradient info during back propagation

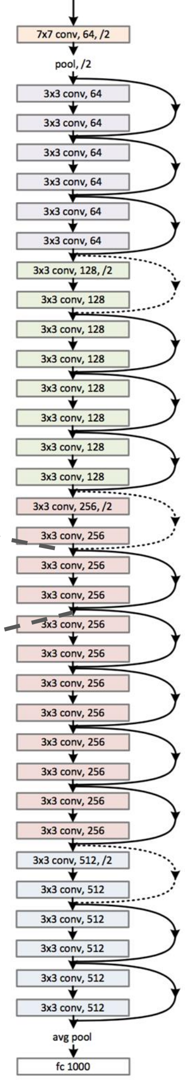
There are many ways to “keep the gradient going”



# Solution

Connect the layers, create a gradient highway or information highway.




ResNet (2015)

Image credit: He et al. (2015)

# Initialization

Can we initialize all neurons to zero?

If all the weights are same we will not be able to break symmetry of the network and all filters will end up learning the same thing.

Large numbers, might knock relu units out.

Relu units once knocked out and their output is zero, their gradient flow also becomes zero.

We need small random numbers at initialization.

Variance : 1/sqrt(n)
Mean: 0

**Popular initialization setups**

(Xavier, Kaiming) (Uniform, Normal)

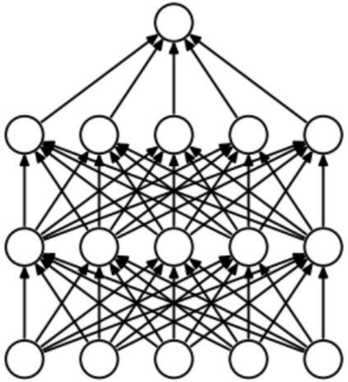
# Dropout

What does cutting off some network connections do?

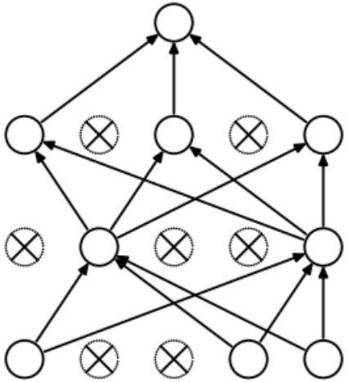
Trains multiple smaller networks in an ensemble.

Can drop entire layer too!

Acts like a really good regularizer



(a) Standard Neural Net



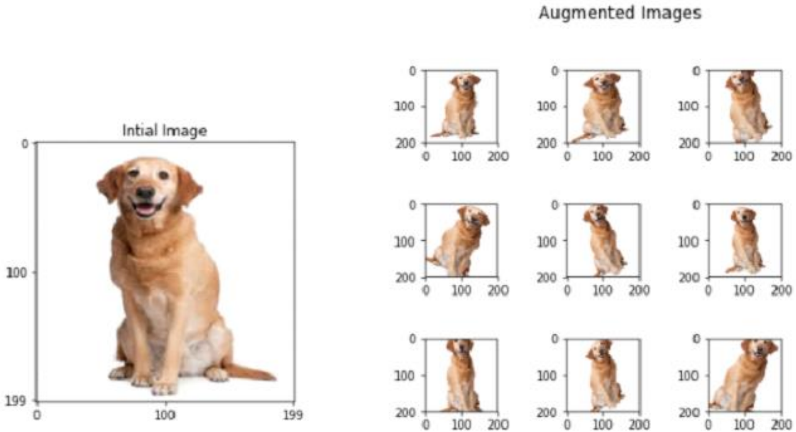
(b) After applying dropout.

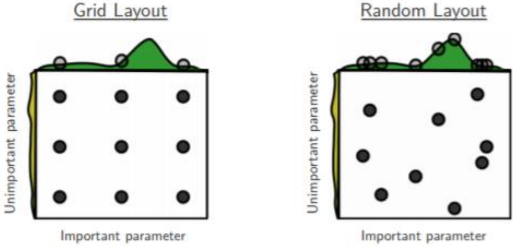
# Tricks for training

Data augmentation if your data set is smaller. This helps the network generalize more.

Early stopping if training loss goes above validation loss.

Random hyperparameter search or grid search?



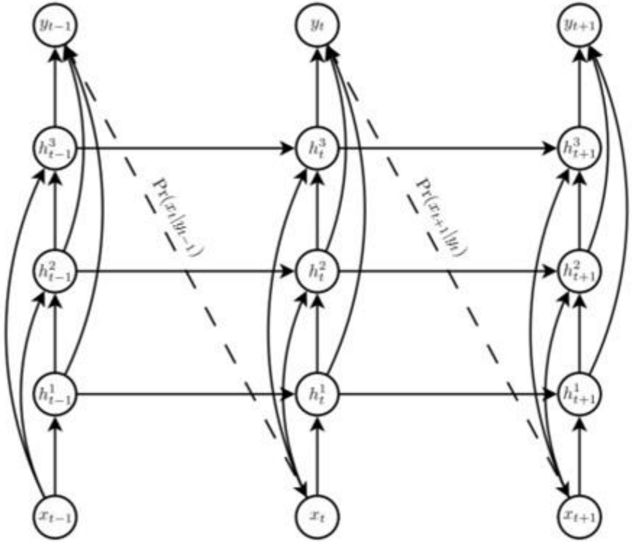



# Overview

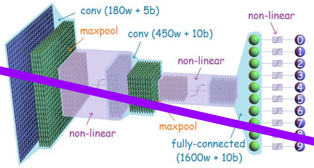
- Motivation for deep learning
- Areas of Deep Learning
- Convolutional neural networks
- Recurrent neural networks
- Deep learning tools

# CNN sounds like fun!
# What are some other areas of deep learning?

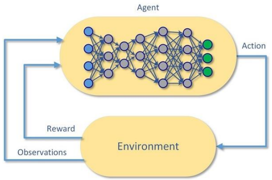
Recurrent NN
Time Series



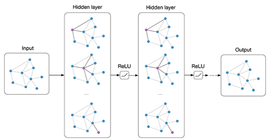

~~Convolutional NN~~




Deep RL




Graph NN

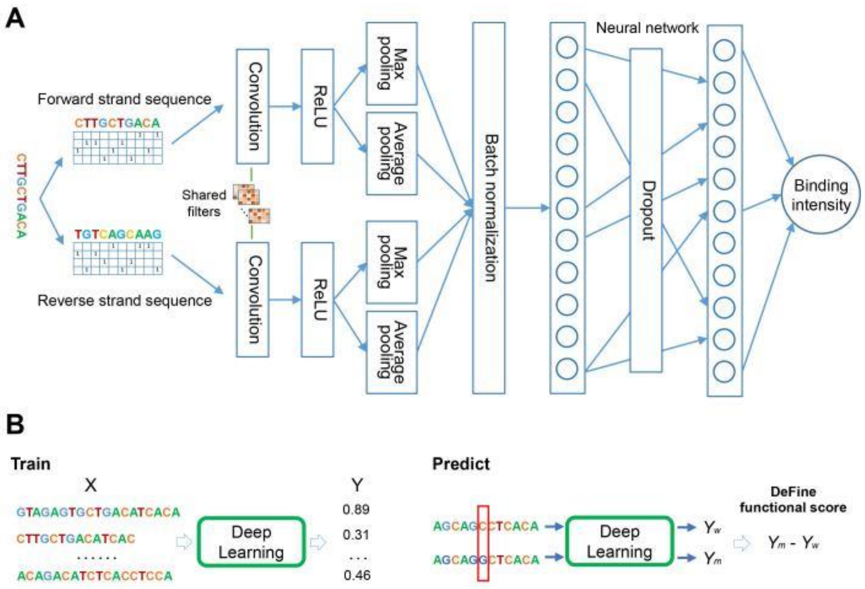


# We can also have 1D architectures (remember this)

CNN works on any data where there is a local pattern

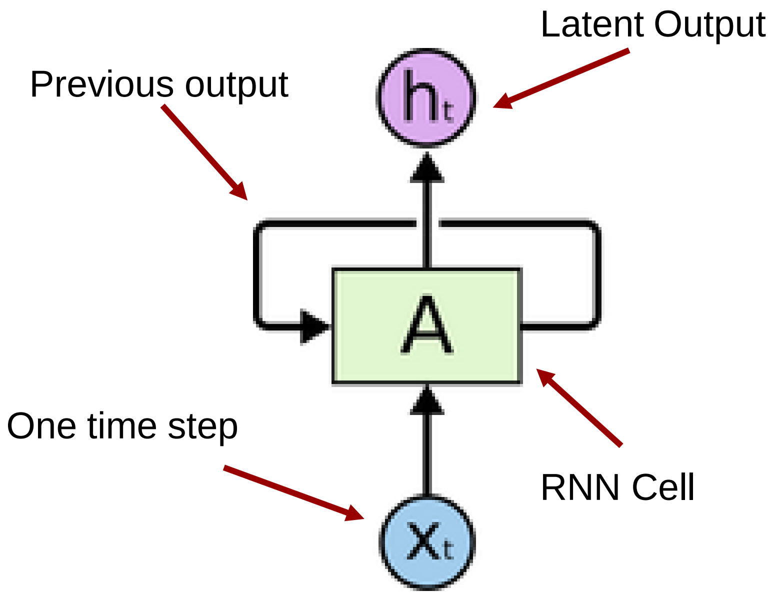
We use 1D convolutions on DNA sequences, text sequences and music notes

But what if time series has **causal dependency** or any kind of **sequential dependency**?

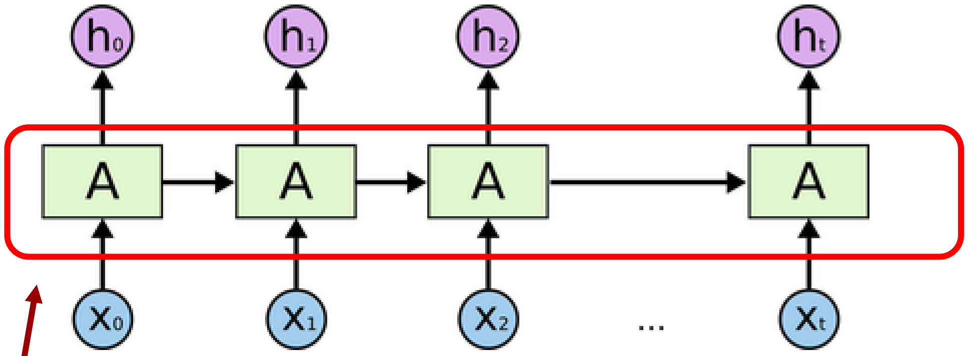


# To address sequential dependency?

Use recurrent neural network (RNN)

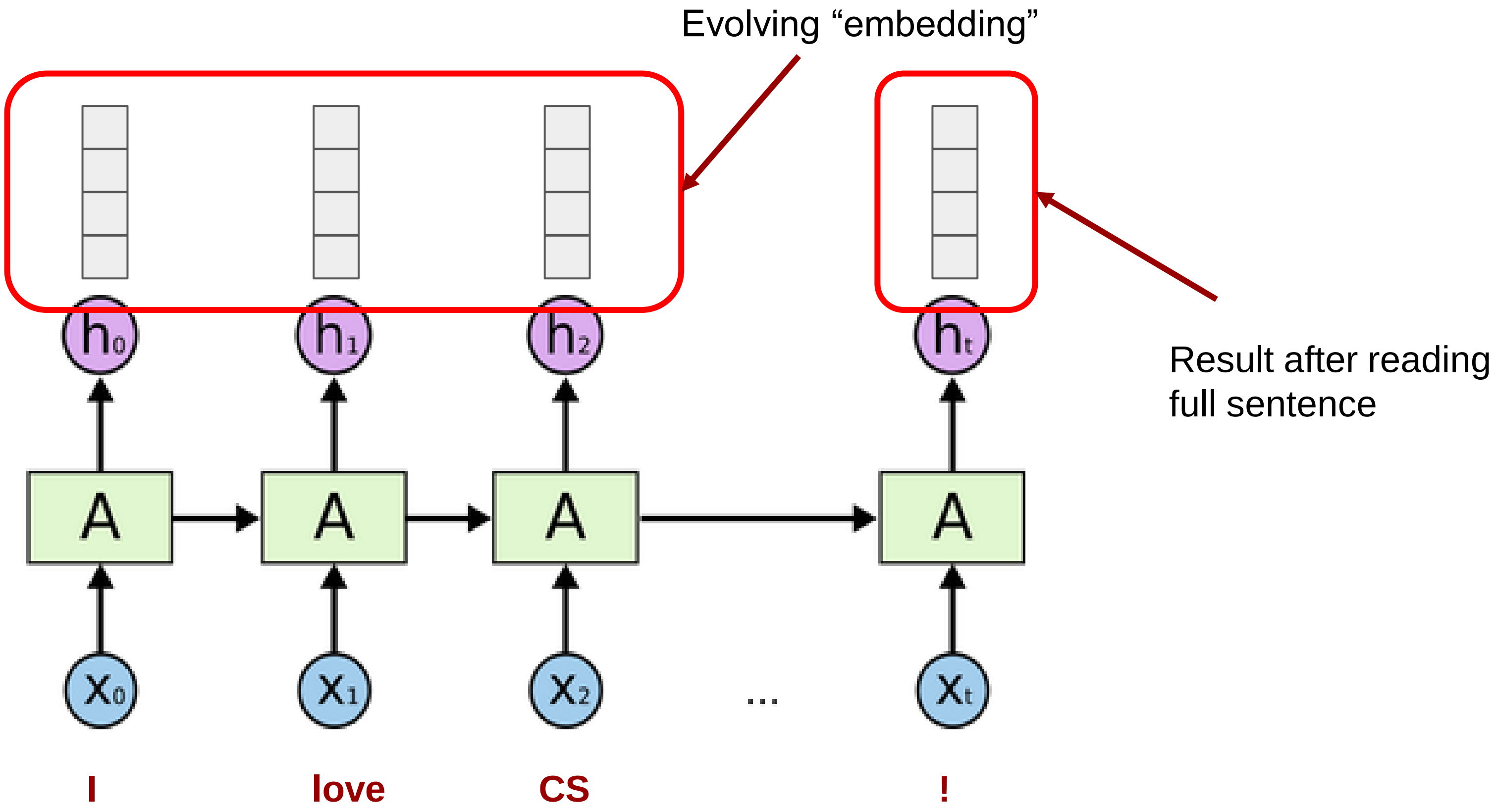



Unrolling an RNN




They are really the same cell,
NOT many different cells like kernels of CNN

# How does RNN produce result?



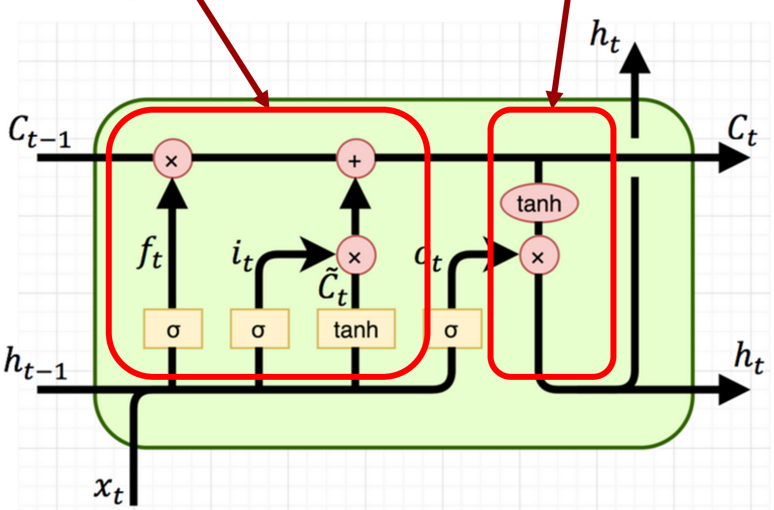
# There are 2 types of RNN cells

Store in “long term memory”

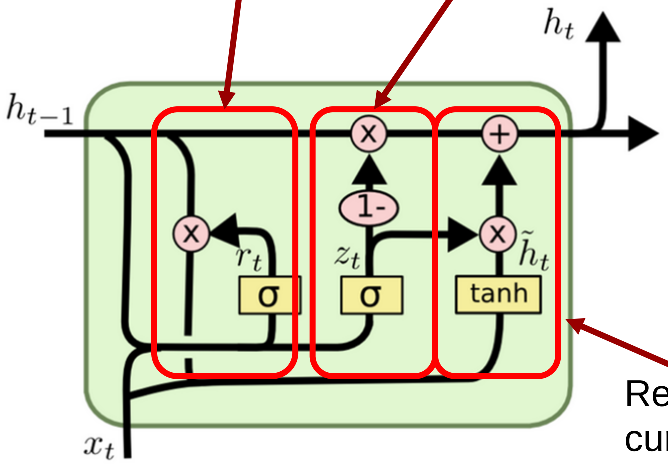
Response to current input

Reset gate

Update gate




Long Short Term Memory (LSTM)

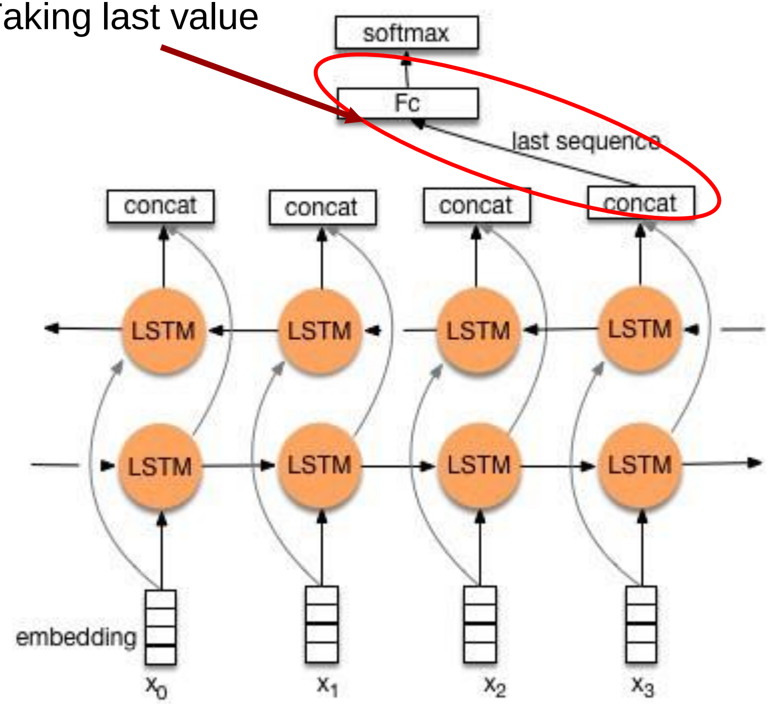



Response to current input

Gated Recurrent Unit (GRU)

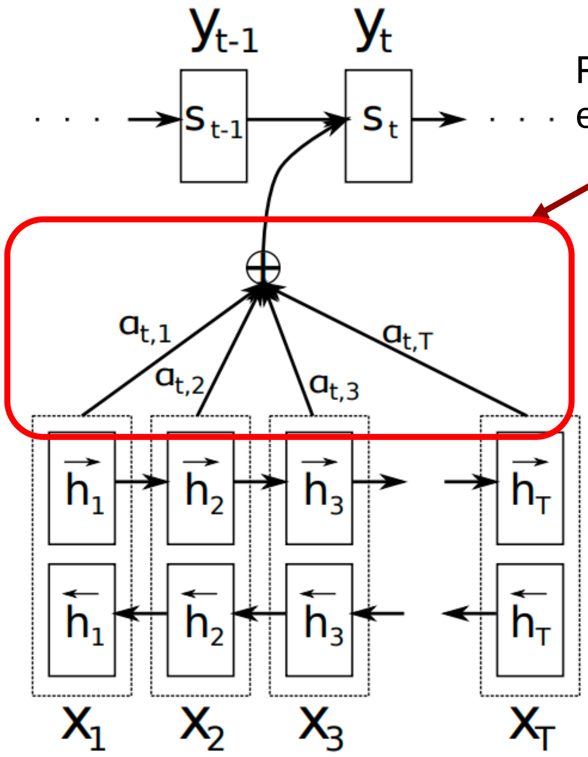
# Recurrent AND deep?

Taking last value




Stacking

Pay “attention” to everything




Attention Model

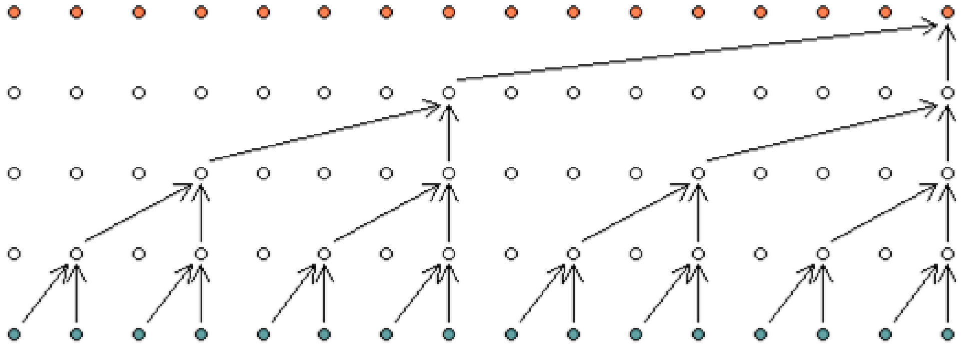
# “Recurrent” AND convolutional?

<u>Temporal convolutional network</u>

Temporal dependency achieved through “one-sided” convolution

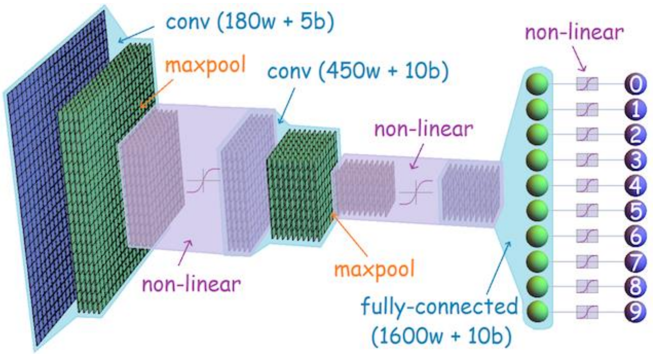
More efficient because deep learning packages are optimized for matrix multiplication = convolution

No hard dependency

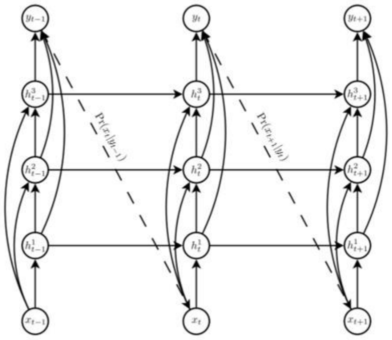


# More? Take CS230, CS236, CS231N, CS224N

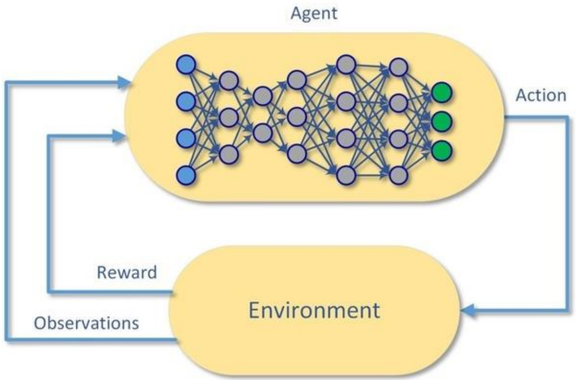
Convolutional NN
Image



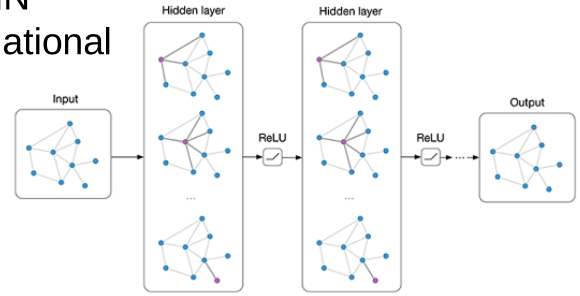

Recurrent NN
Time Series



Deep RL
Control System

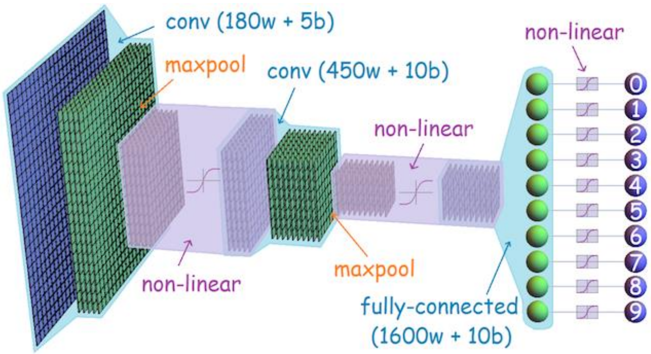



Graph NN
Networks/Relational

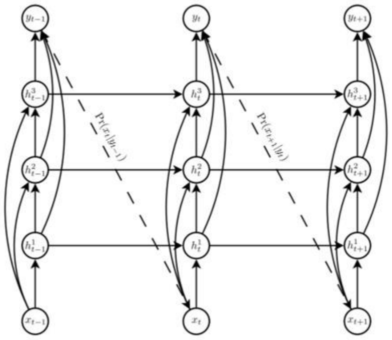


# Not today, but take CS234 and CS224W

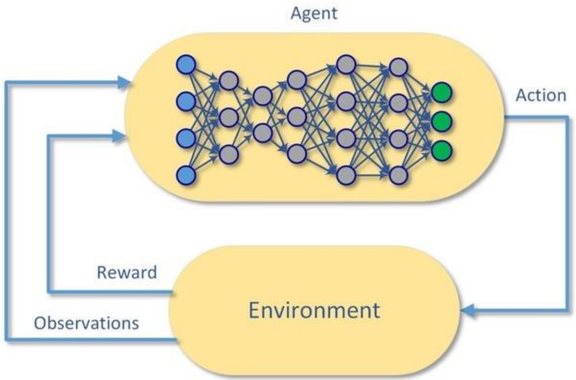
Convolutional NN
Image



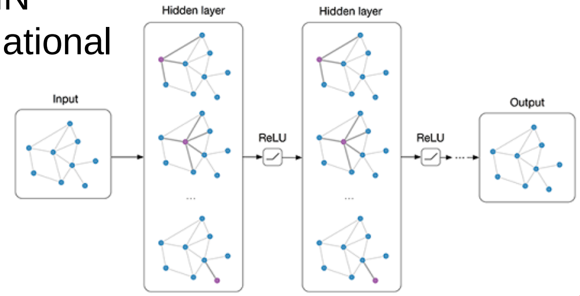

Recurrent NN
Time Series



Deep RL
Control System




Graph NN
Networks/Relational



# Overview

- Motivation for deep learning
- Areas of Deep Learning
- Convolutional neural networks
- Recurrent neural networks
- Deep learning tools

# Tools for deep learning

Keras

TensorFlow

theano

PYTORCH

Popular Tools

Specialized Groups

Caffe2

mxnet

Microsoft CNTK

# $50 not enough! Where can I get free stuff?

Google Colab

Free (limited-ish) GPU access

Works nicely with Tensorflow

Links to Google Drive

Azure Notebook

Kaggle kernel???

Amazon SageMaker?

Register a new Google Cloud account

=> Instant $300??

=> AWS free tier (limited compute)

=> Azure education account, $200?

To **SAVE** money

**CLOSE** your GPU instance

**~$1** an hour

Good luck!
Well, have fun too :D