Introduction to Deep Learning

Angelica Sun
(adapted from Atharva Parulekar, Jingbo Yang)

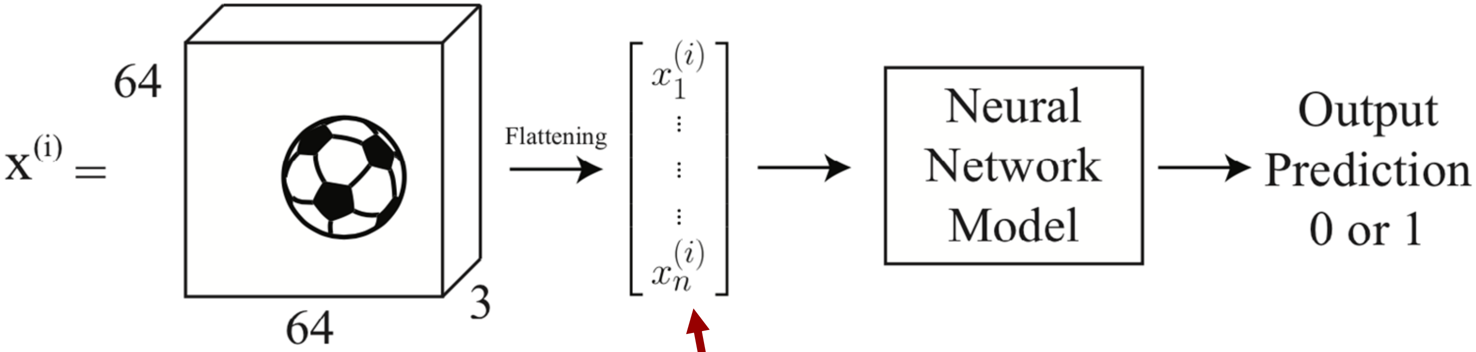
Overview

- Motivation for deep learning
- Convolutional neural networks
- Recurrent neural networks
- Transformers
- Deep learning tools

But we learned multi-layer perceptron in class?

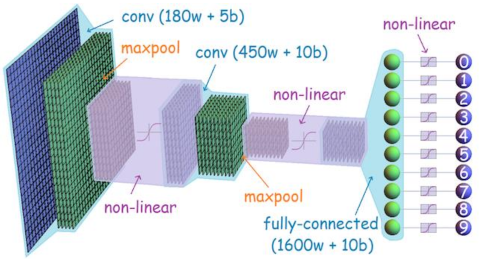
Expensive to learn. Will not generalize well.

Does not exploit the <u>order and local relations</u> in the data!

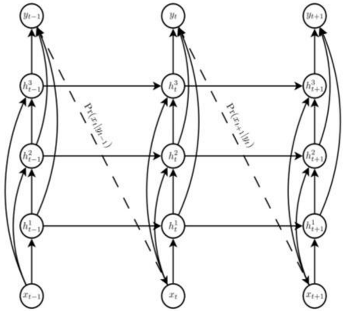


64x64x3=12288 parameters
We also want **many** layers

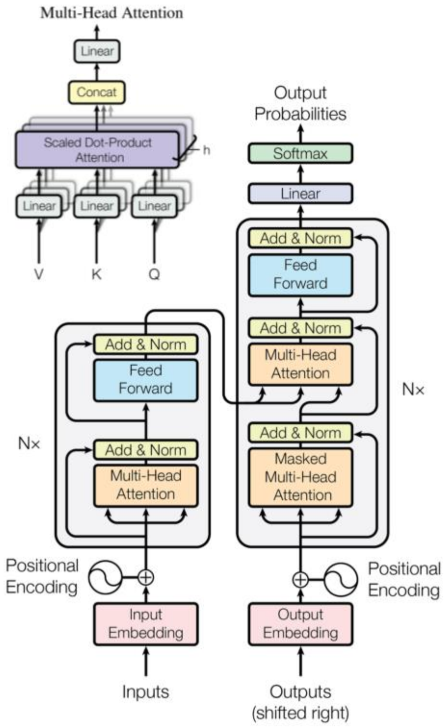
What are areas of deep learning?



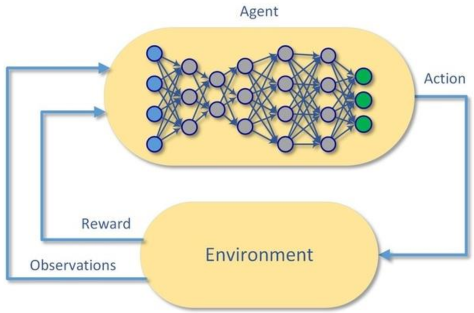
Convolutional NN
Image



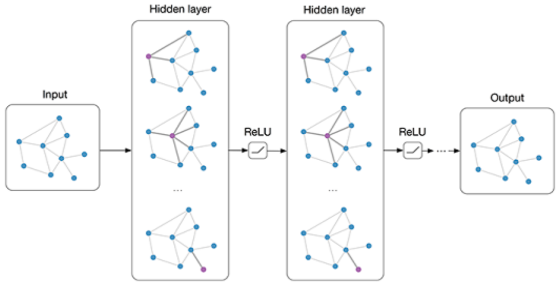
Recurrent NN
Sequential Inputs



Transformers
Parallelized
Sequential Inputs



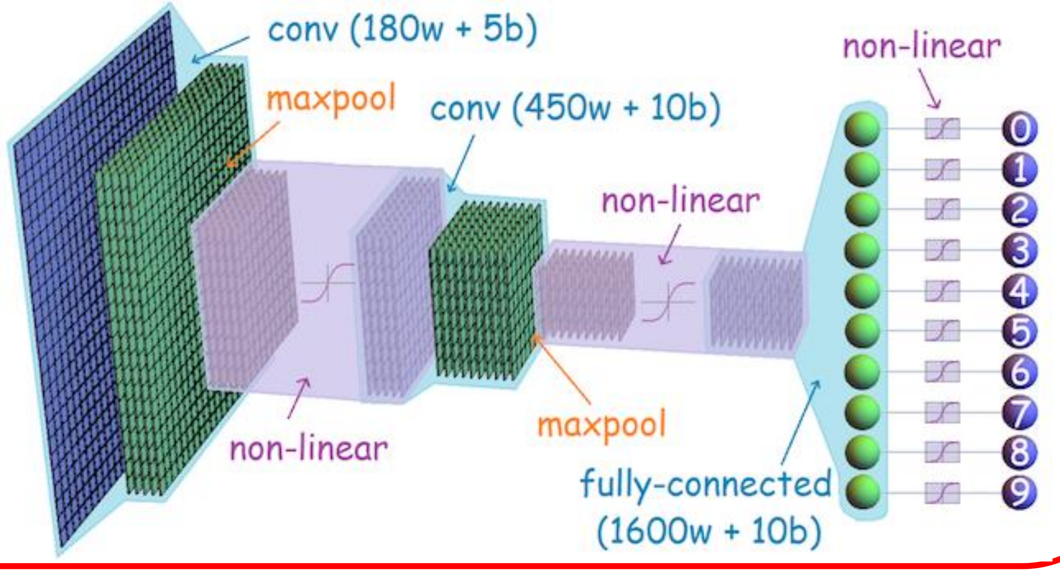
Deep RL
Control System



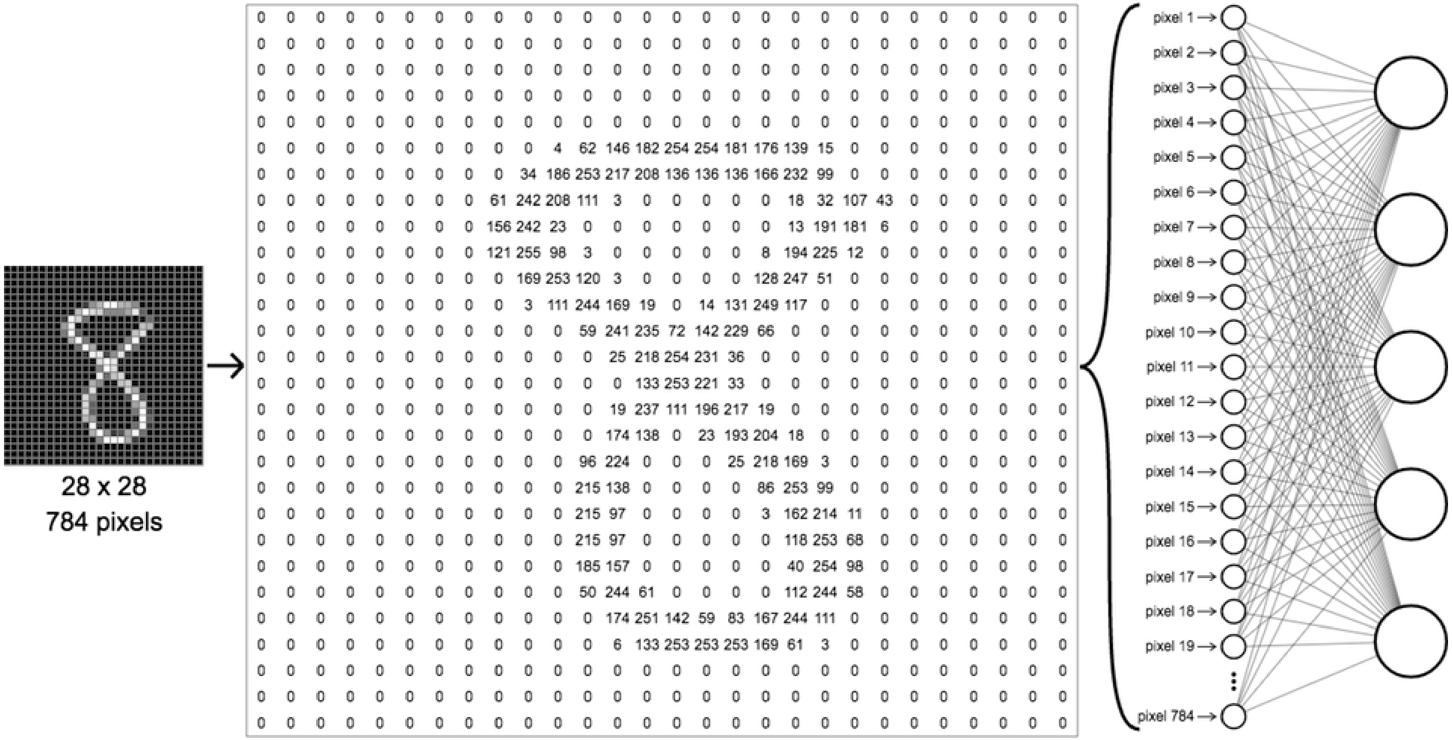
Graph NN
Networks/Relational

Starting from CNN

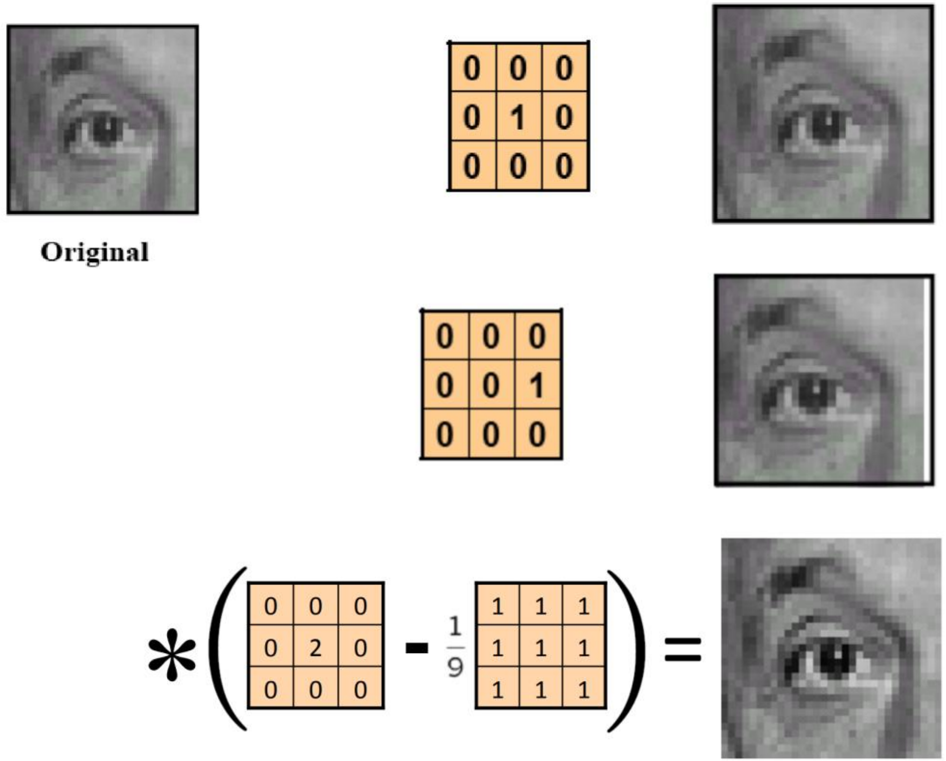
Convolutional Neural Network



Let us look at images in detail



Filters in traditional Computer Vision



Sharpening filter
(accentuates edges)

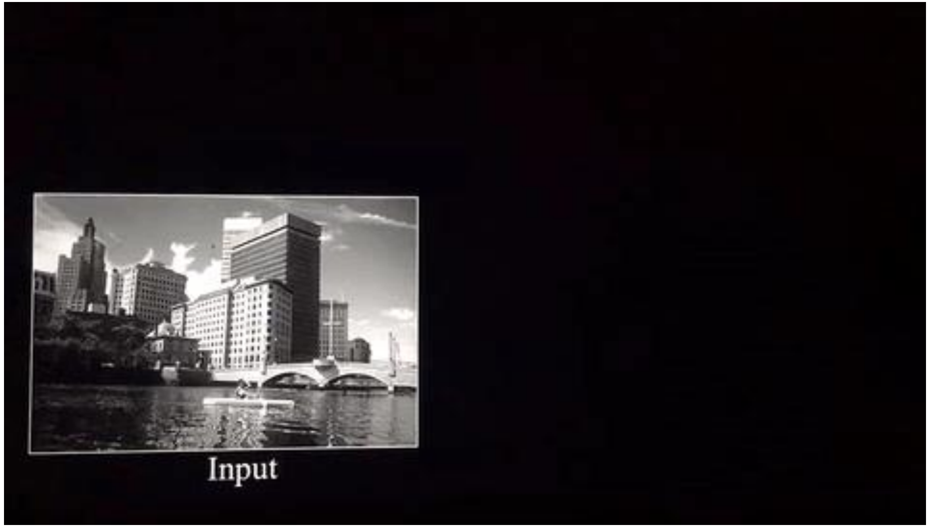
Image credit:
https://home.ttic.edu/~rurtasun/courses/CV/lecture02.pdf

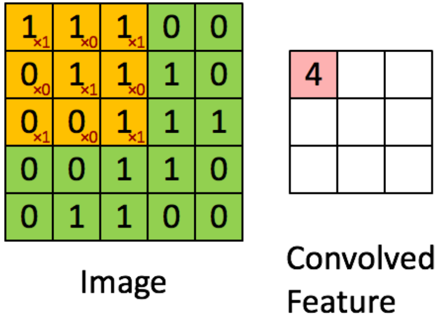
Learning filters in CNN

Why not extract features using filters?

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

Learnable filters!!





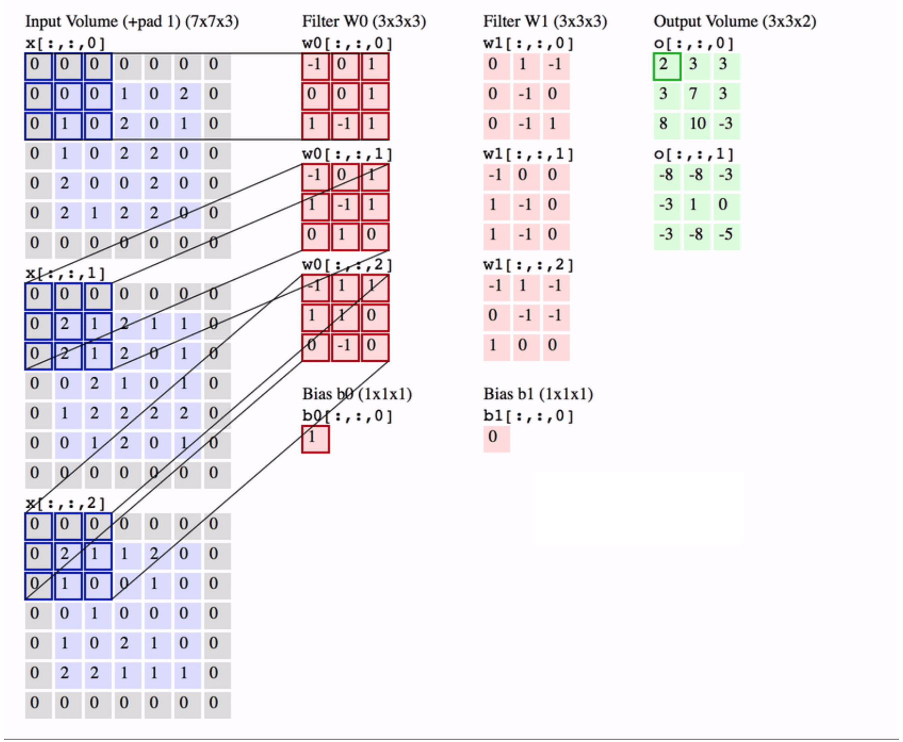
Convolution on multiple channels

Images are generally RGB !!

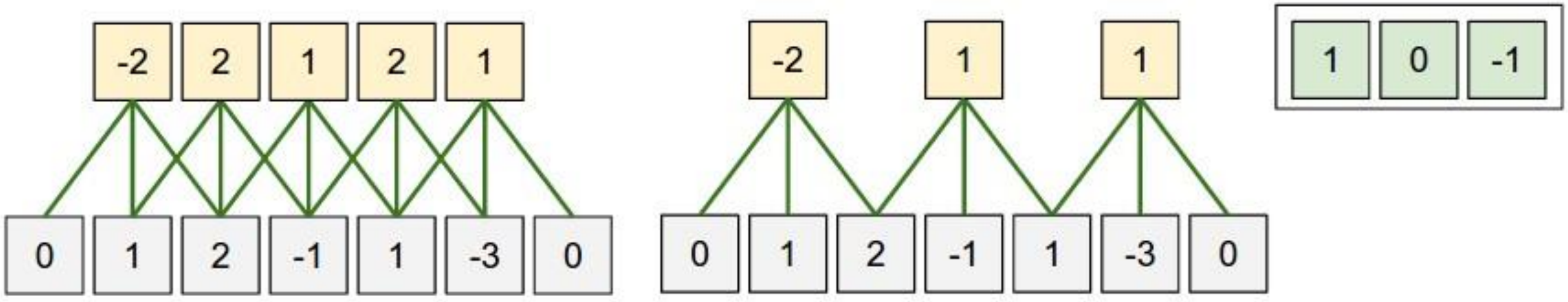
How would a filter work on a image with RGB channels?

The filter should also have 3 channels.

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



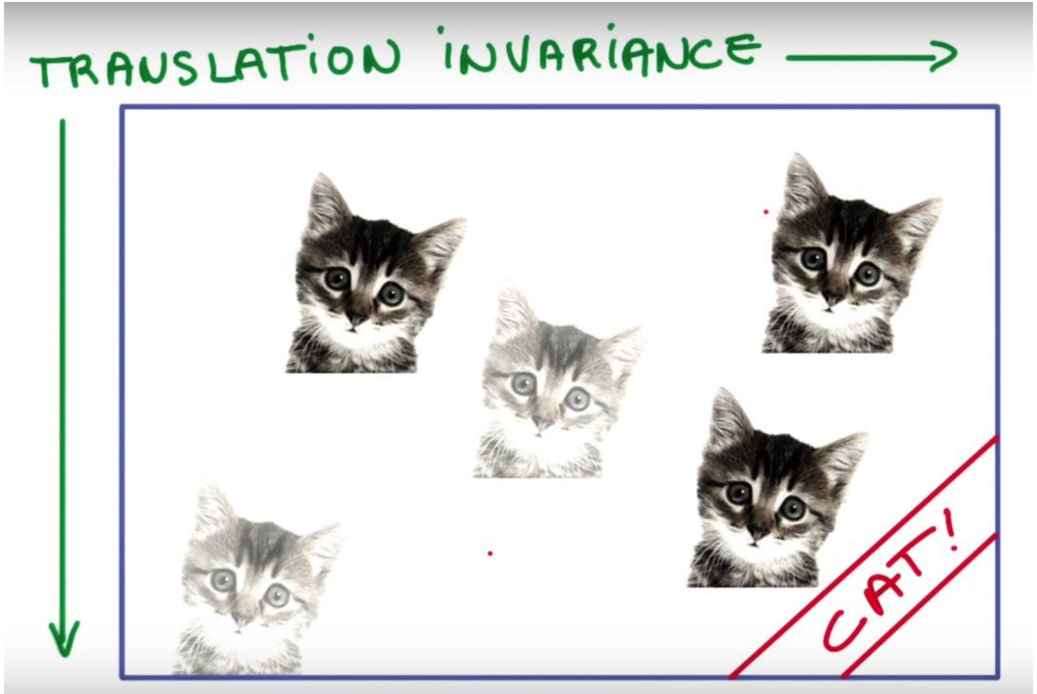
Parameter Sharing



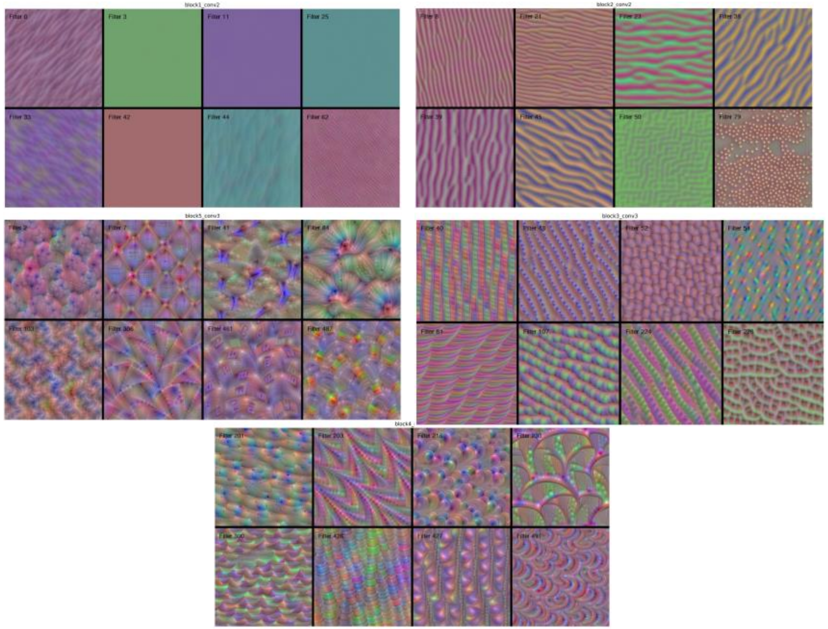
Lesser the parameters less computationally intensive the training. This is a win win as we are reusing parameters.

Translational invariance

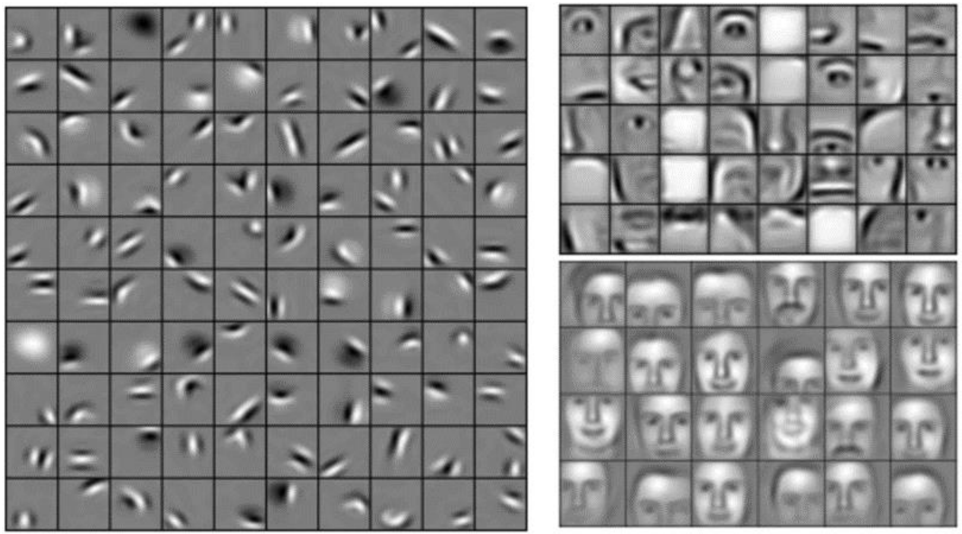
Since we are training filters to detect cats and the moving these filters over the data, a differently positioned cat will also get detected by the same set of filters.



Visualizing learned 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.

A typical CNN structure:

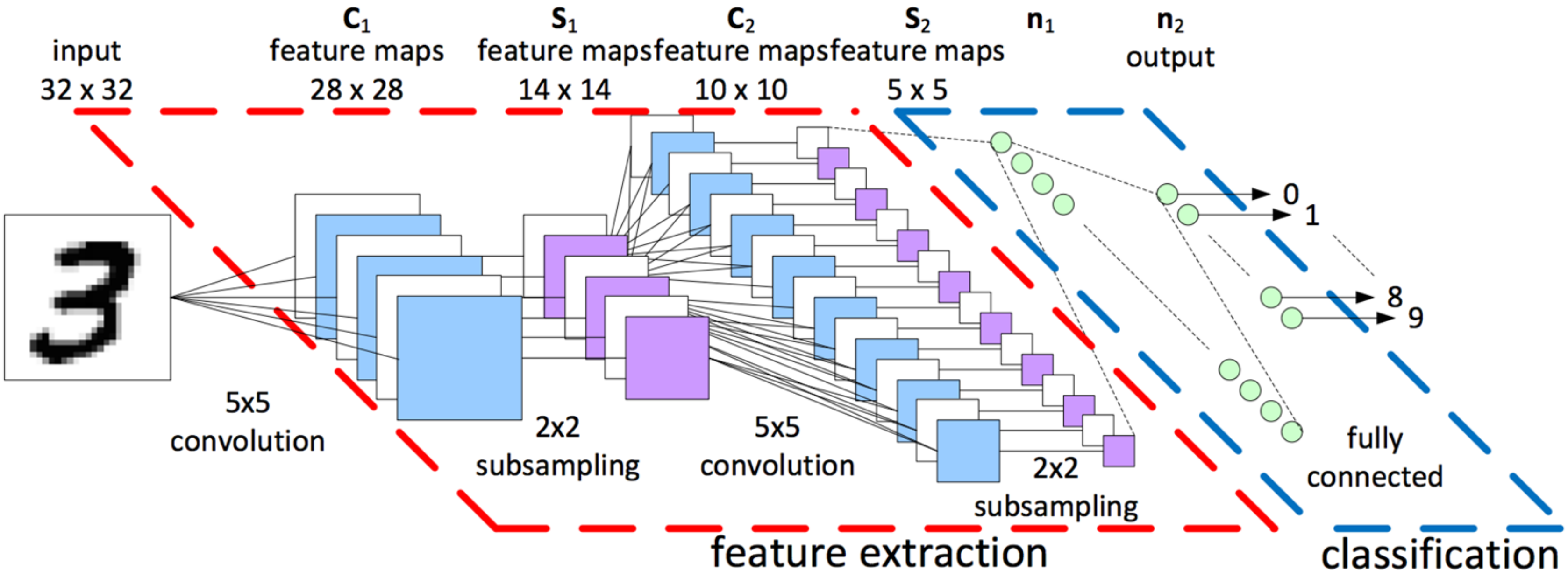


Image credit: LeCun et al. (1998)

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.

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

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

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

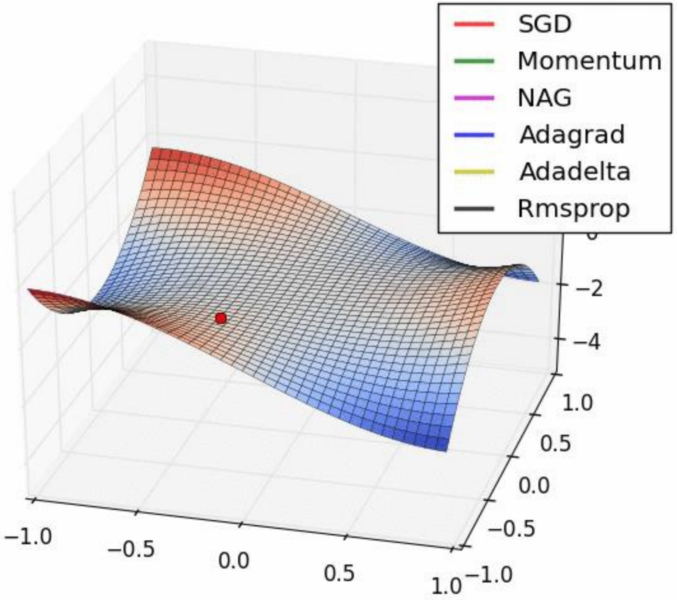
SOTA Example – Detectron2

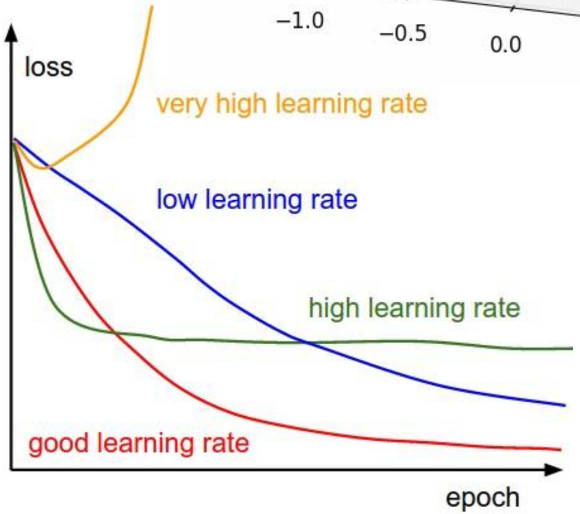
How do we learn?

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

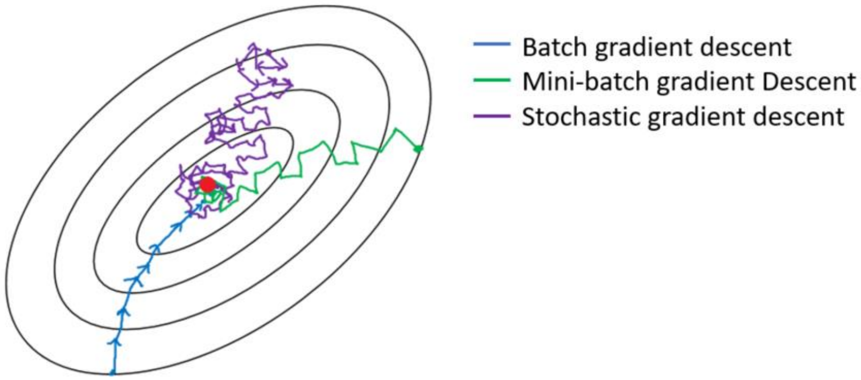
Expensive to compute gradient for large dataset

Memory size

Compute time

Mini-batch: takes a sample of training data

How to we sample intelligently?

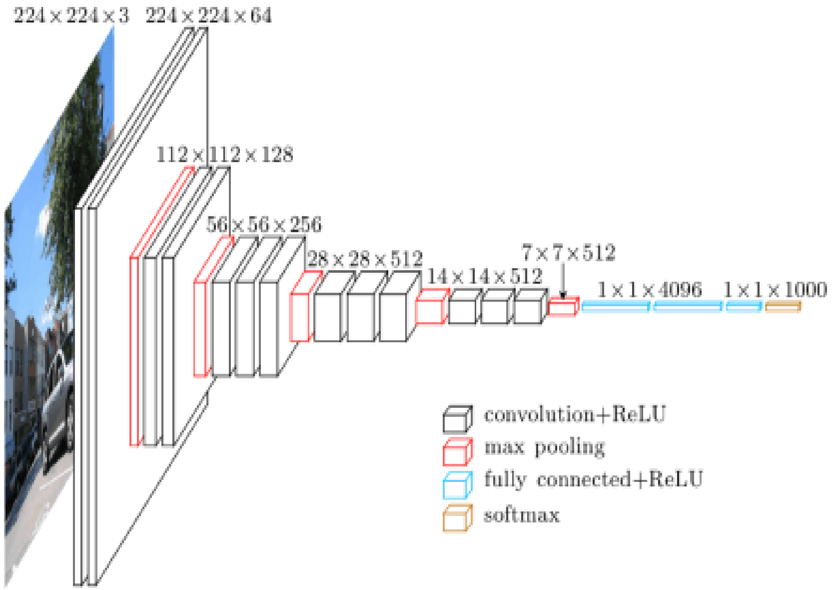


Is deeper better?

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

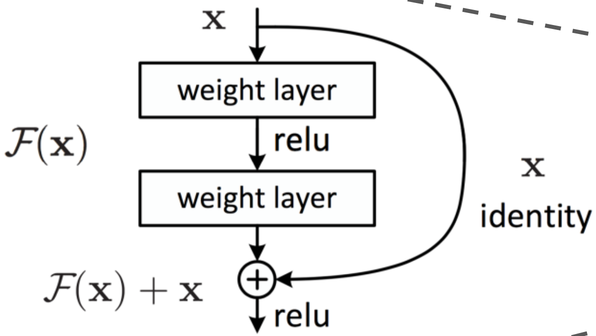
- Loss of information during forward propagation
- Loss of gradient info during back propagation

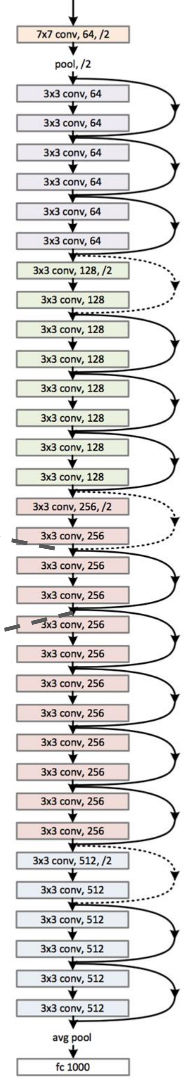
There are many ways to “keep the gradient going”



One Solution: skip connection

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)

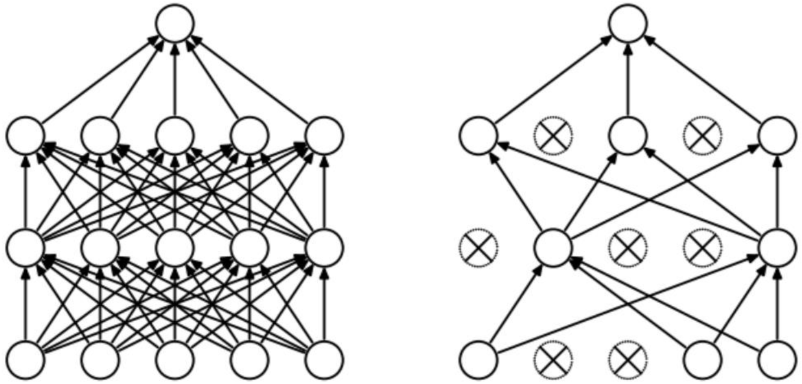
Dropout

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



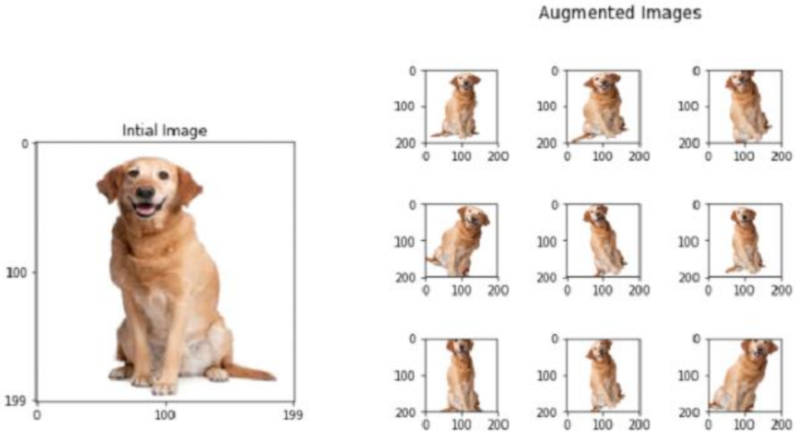
(a) Standard Neural Net (b) After applying dropout.

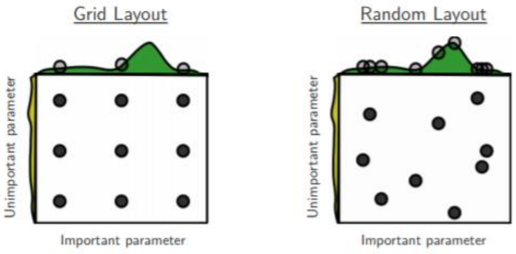
More tricks for training

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

Early stopping if training loss goes above validation loss.

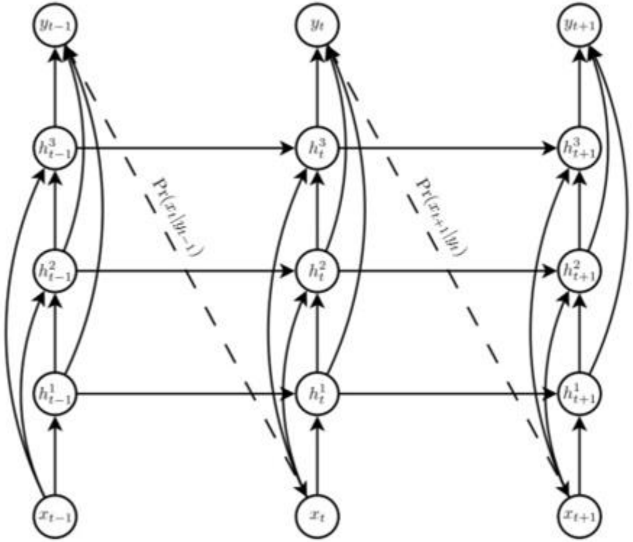
Random hyperparameter search or grid search?



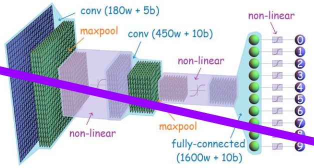


CNN sounds like fun!
What are some other areas of deep learning?

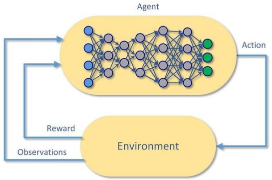
Recurrent NN
Sequential data



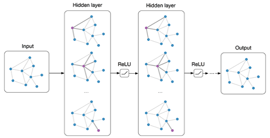
~~Convolutional NN~~



Deep RL



Graph NN

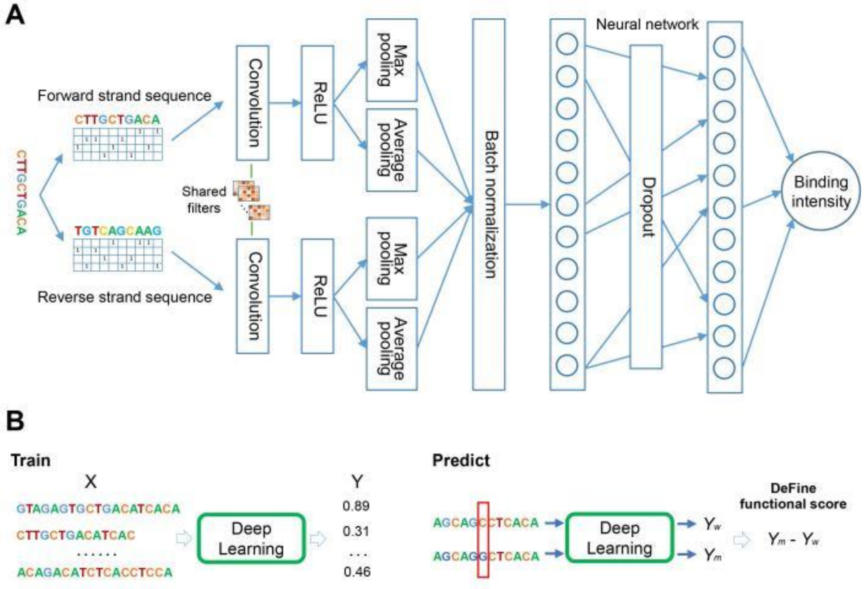


We can also have 1D architectures (remember this)

CNN works on any data where there is a local pattern

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

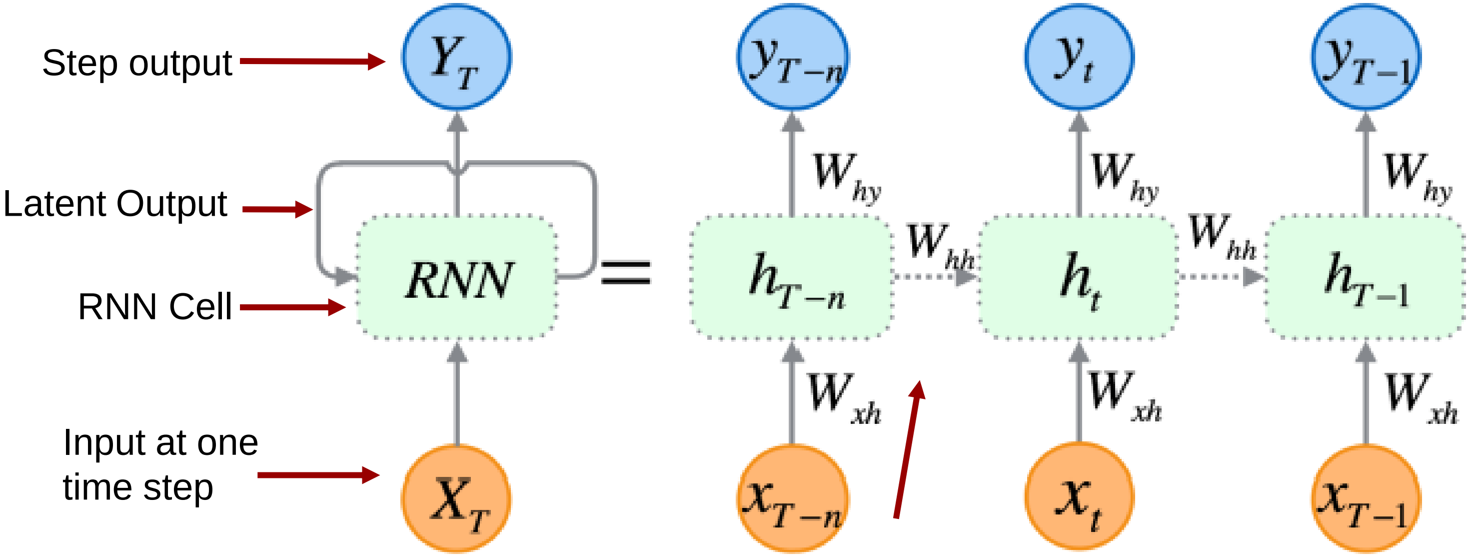
But what if time series has **causal dependency** or any kind of **sequential dependency**?



To address sequential dependency?

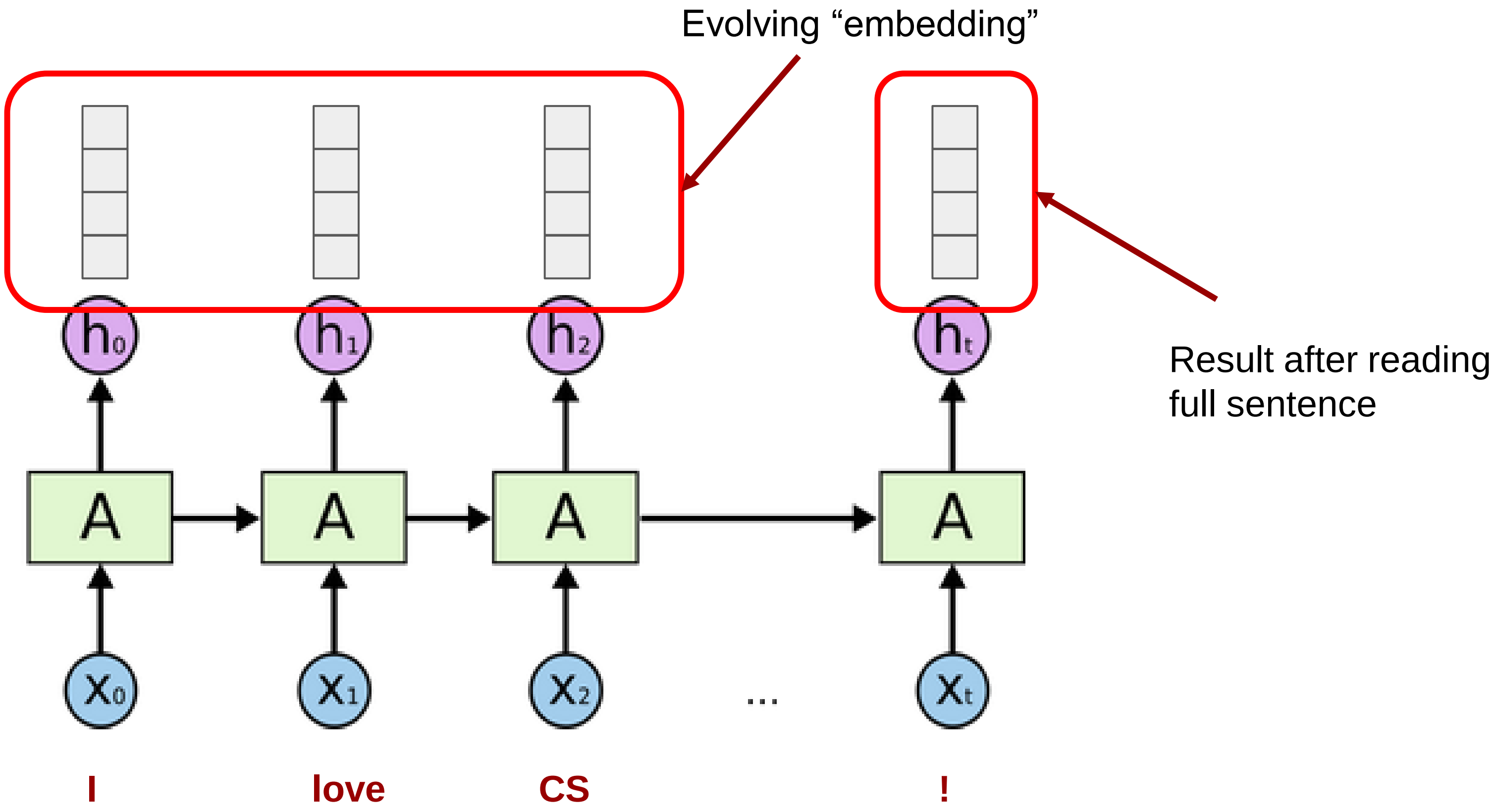
Use recurrent neural network (RNN)

Unrolling an RNN

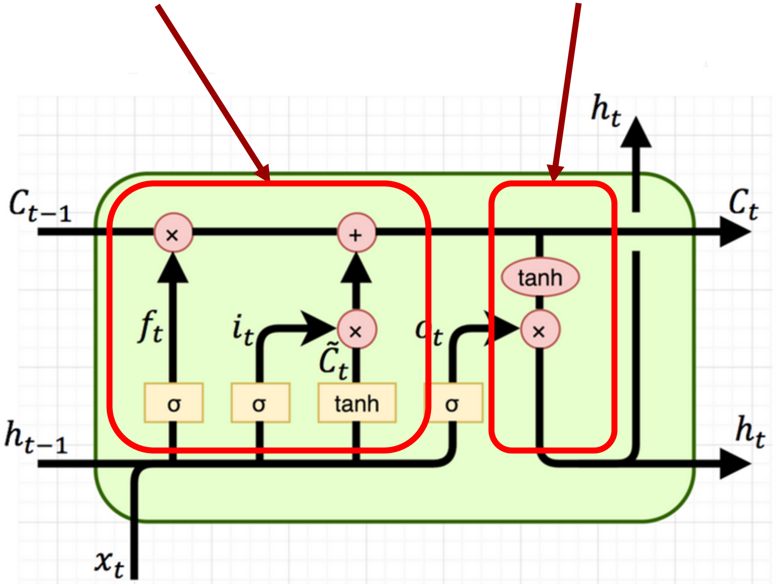


The RNN Cell (Composed of Wxh and Whh in this example) is really the same cell.
NOT many different cells like the filters of CNN.

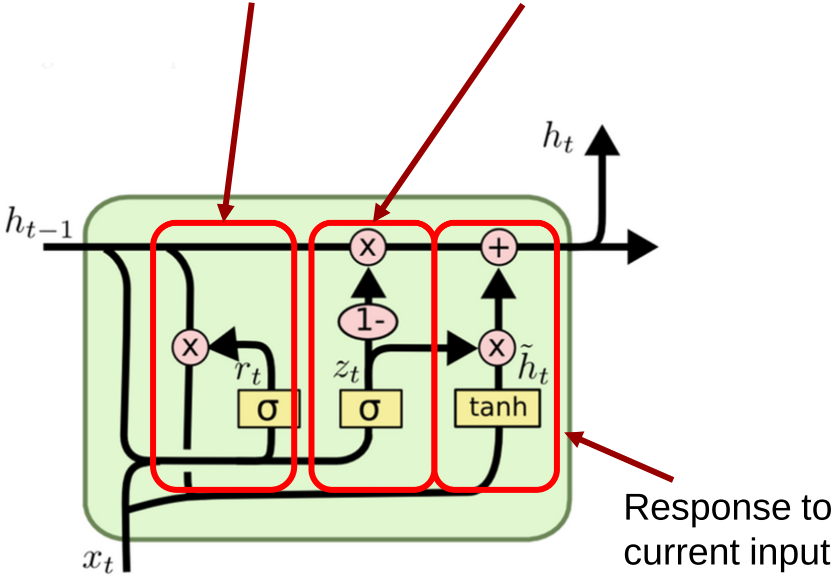
How does RNN produce result?



2 Typical RNN Cells



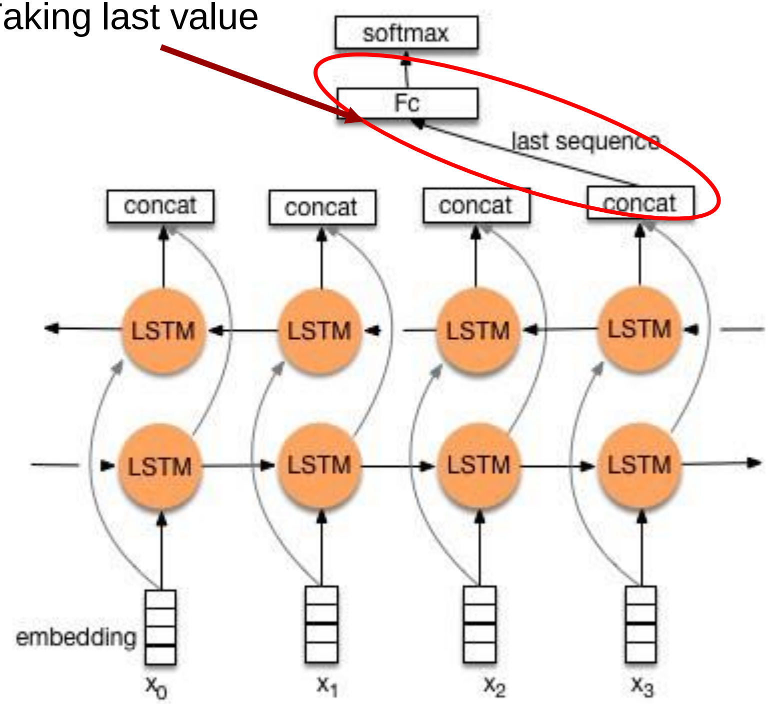
Long Short Term Memory (**LSTM**)



Gated Recurrent Unit (**GRU**)

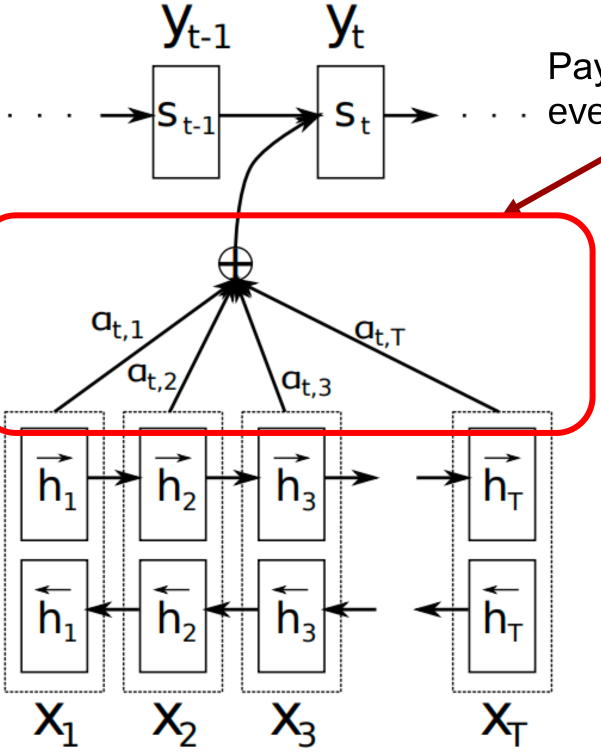
Recurrent AND deep?

Taking last value



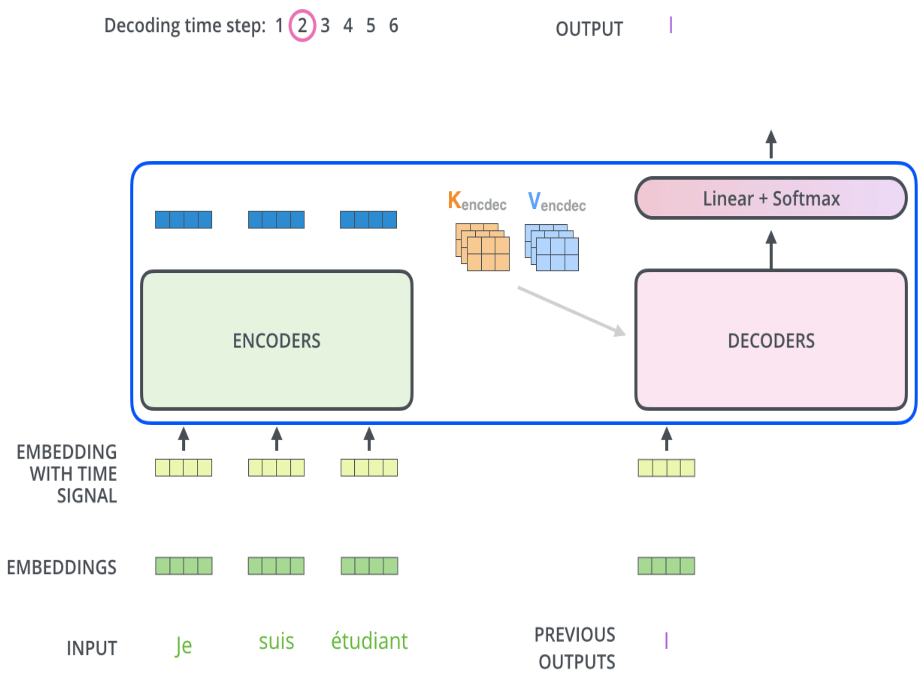
Stacking

Pay “attention” to everything



Attention Model

Transformer – Attention is All You Need!

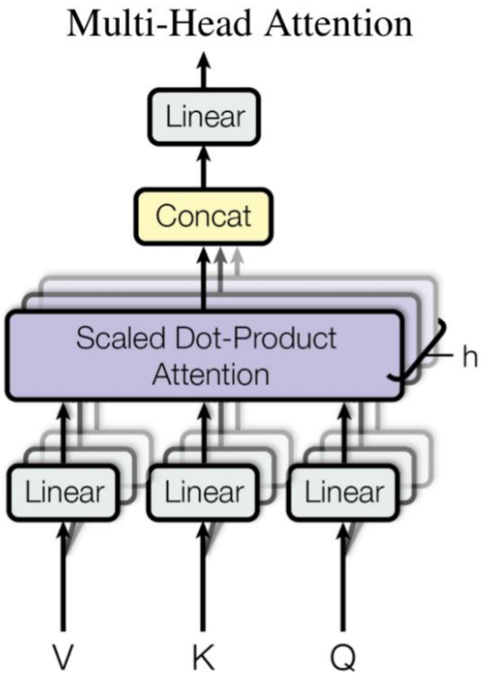


Originally proposed for translation.

Encoder computes hidden representations for each word in the input sentence
Applies **self attention.**

Decoder makes sequential prediction similar as in RNN
At each time step, it predicts the next word based on its previous predictions (partial sentence).
Applies **self attention** and **attention on encoder** outputs.

Transformer – Attention is All You Need!

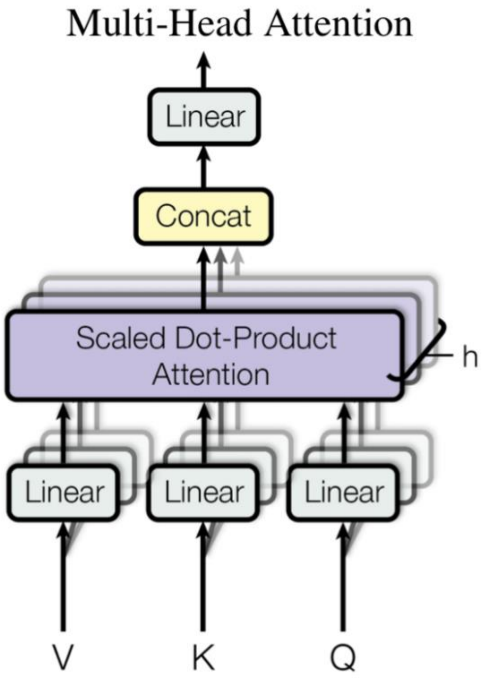


The dot product in softmax below computes how each word of sequence 1 (Q) is influenced by all the other words in the sequence 2 (K).

Considering the different importance, we computed a weighted sum of the information in the sequence 2 (V) to use in computing the hidden representation of sequence 1.

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Transformer – Attention is All You Need!

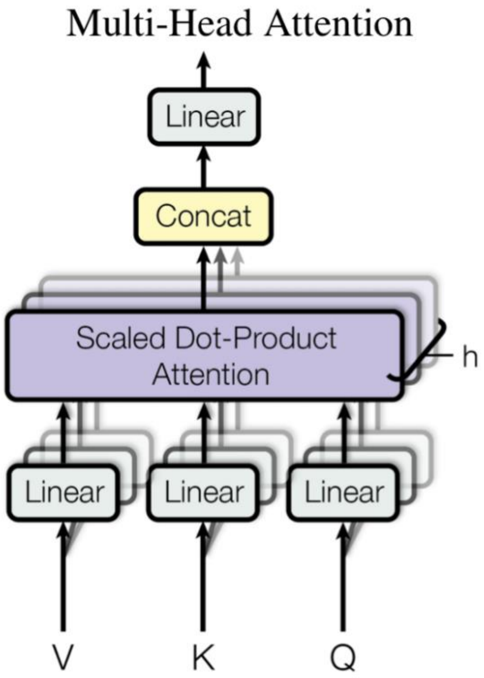


The dot product in softmax below computes how each word of sequence 1 (Q) is influenced by all the other words in the sequence 2 (K).

Considering the different importance, we computed a weighted sum of the information in the sequence 2 (V) to use in computing the hidden representation of sequence 1.

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Transformer – Attention is All You Need!

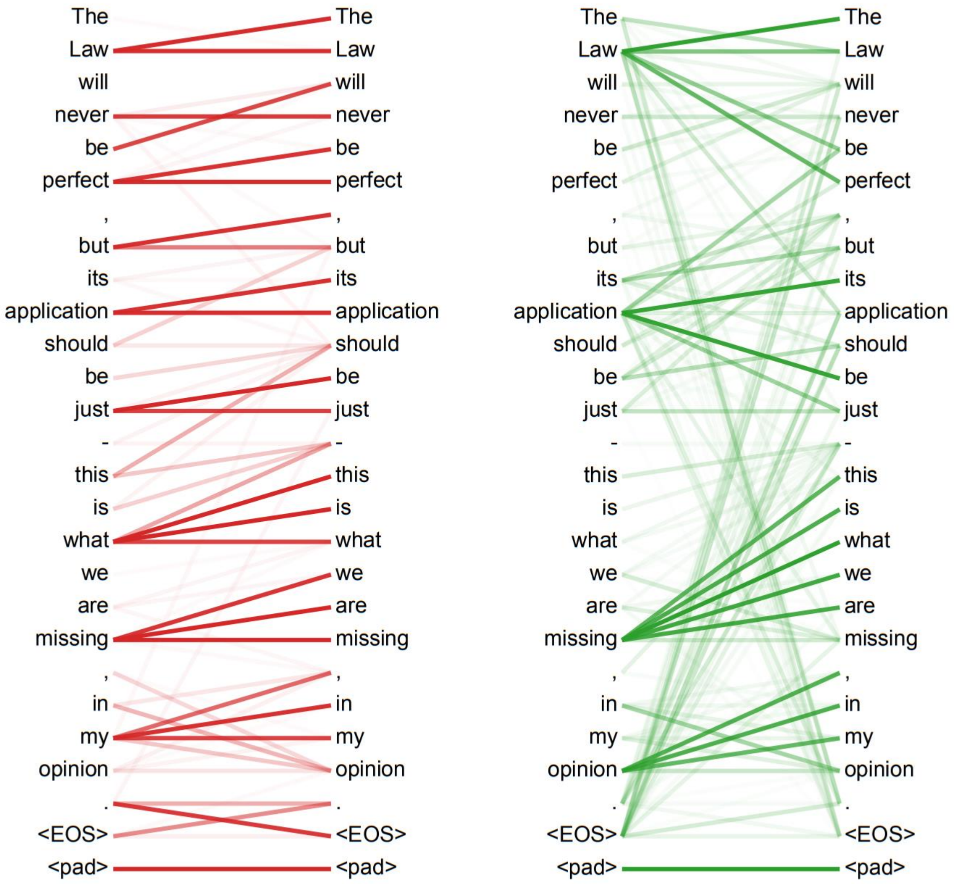


Multiple heads!

-- Similar as how you have multiple filters in CNN

Loss of sequential order?

-- Positional encoding! (often use sine waves)



Examples of attention scores from two different self-attention heads.

References:

https://arxiv.org/pdf/1706.03762.pdf
https://medium.com/inside-machine-learning/what-is-a-transformer-d07dd1fbec04
https://towardsdatascience.com/transformers-141e32e69591
https://towardsdatascience.com/transformers-explained-visually-part-2-how-it-works-step-by-step-b49fa4a64f34

SOTA Example – GPT3

A "whatpu" is a small, furry animal native to Tanzania. An example of a sentence that uses the word whatpu is:
We were traveling in Africa and we saw these very cute whatpus.

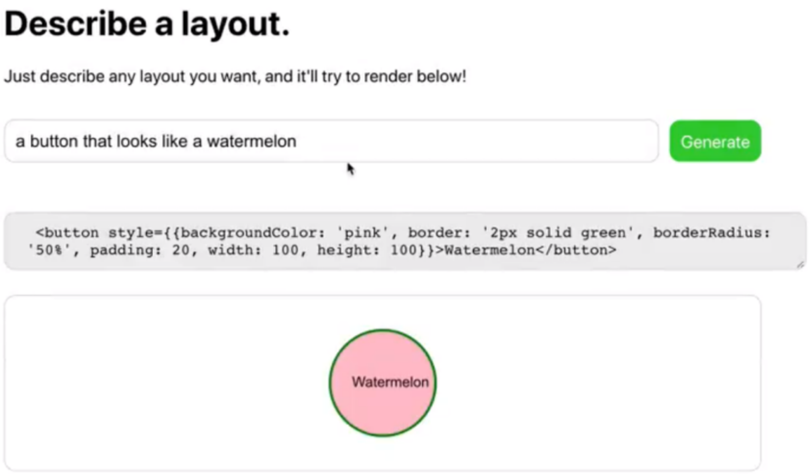
To do a "farduddle" means to jump up and down really fast. An example of a sentence that uses the word farduddle is:
One day when I was playing tag with my little sister, she got really excited and she started doing these crazy farduddles.
A "yalubalu" is a type of vegetable that looks like a big pumpkin. An example of a sentence that uses the word yalubalu is:
I was on a trip to Africa and I tried this yalubalu vegetable that was grown in a garden there. It was delicious.

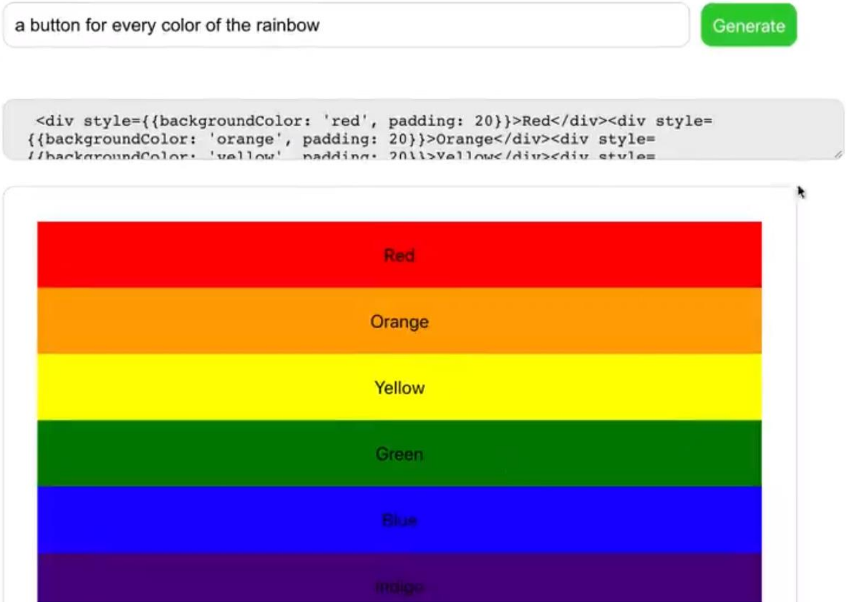
A "Burringo" is a car with very fast acceleration. An example of a sentence that uses the word Burringo is:
In our garage we have a Burringo that my father drives to work every day.

A "Gigamuru" is a type of Japanese musical instrument. An example of a sentence that uses the word Gigamuru is:
I have a Gigamuru that my uncle gave me as a gift. I love to play it at home.

To "screeg" something is to swing a sword at it. An example of a sentence that uses the word screeg is:
We screeghed at each other for several minutes and then we went outside and ate ice cream.

SOTA Example – GPT3





SOTA Example – DALLE

TEXT PROMPT

an illustration of a baby daikon radish in a tutu walking a dog

AI-GENERATED IMAGES

Edit prompt or view more images↓

TEXT PROMPT

an armchair in the shape of an avocado [...]

AI-GENERATED IMAGES

Edit prompt or view more images↓

TEXT PROMPT

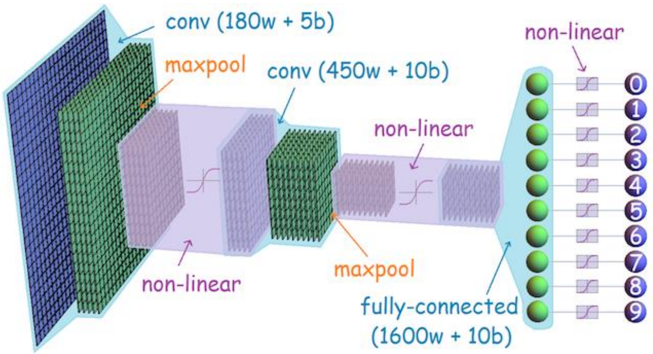
a store front that has the word ‘openai’ written on it [...]

AI-GENERATED IMAGES

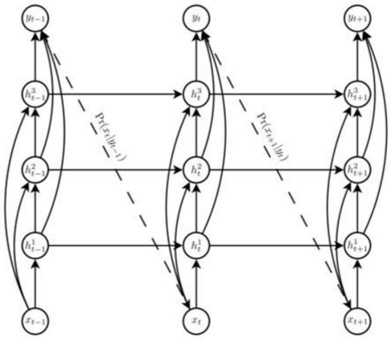
Edit prompt or view more images↓

More? Take CS230, CS236, CS231N, CS224N

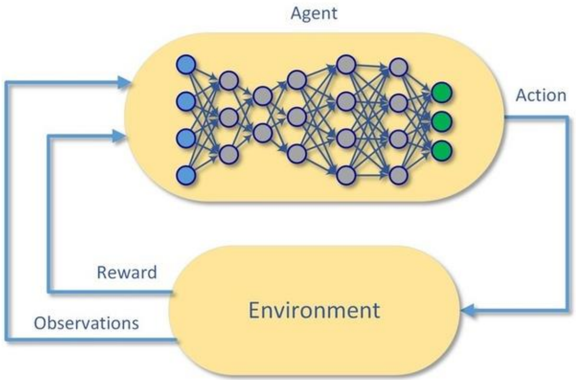
Convolutional NN
Image



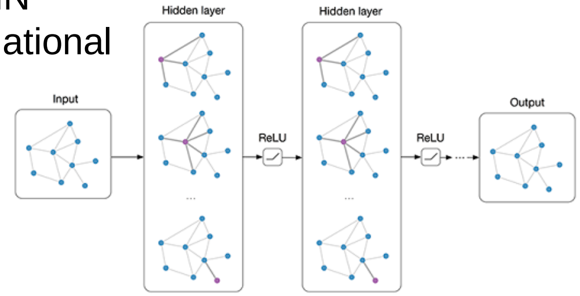
Recurrent NN
Time Series



Deep RL
Control System

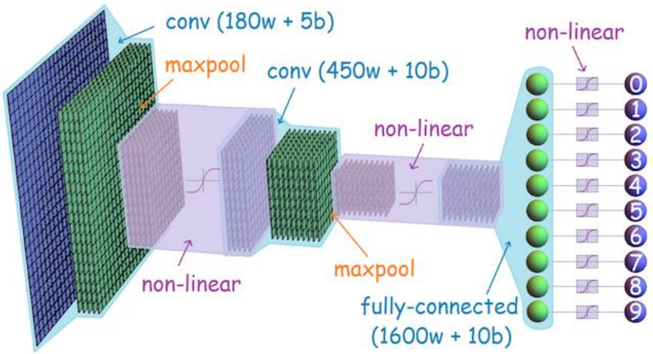


Graph NN
Networks/Relational

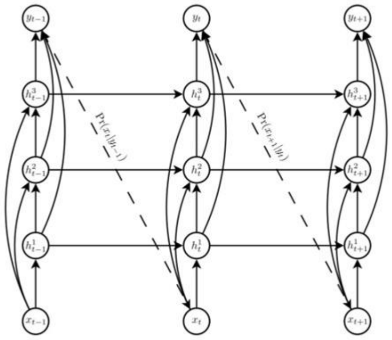


Not today, but take CS234 and CS224W

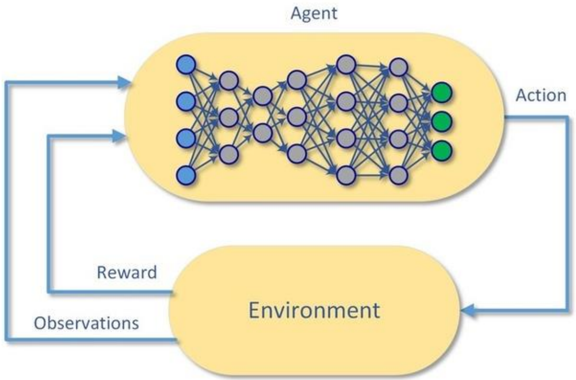
Convolutional NN
Image



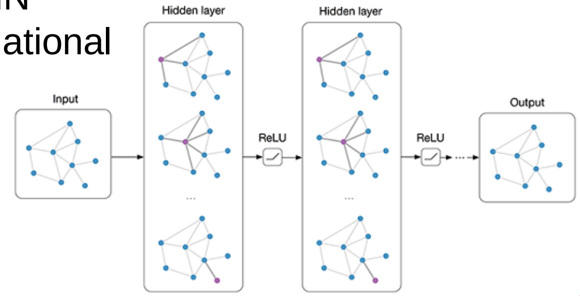
Recurrent NN
Time Series



Deep RL
Control System



Graph NN
Networks/Relational



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