Convolutional Neural Networks

CPSC 433: Artificial Intelligence
Fall 2024

Jonathan Hudson, Ph.D.
Assistant Professor (Teaching)
Department of Computer Science
University of Calgary

August 8, 2024

UNIVERSITY OF CALGARY

Outlines

- ImageNet
- Convolution
- Pooling
- Examples
- Batch Normalization
- Pre-processing
- Reflection on ImageNet
- Other Network Types

ImageNet

Deep Learning/ Convolutional Neural Networks

- Classify an image into 1000 possible classes:
  e.g. Abyssinian cat, Bulldog, French Terrier, Cormorant, Chickadee, red fox, banjo, barbell, hourglass, knot, maze, viaduct, etc.

→

cat, tabby cat (0.71)
Egyptian cat (0.22)
red fox (0.01)
.....

The Data: ILSVRC

- Imagenet Large Scale Visual Recognition Challenge (ILSVRC): Annual Competition
- 2010->2017

1000 Categories

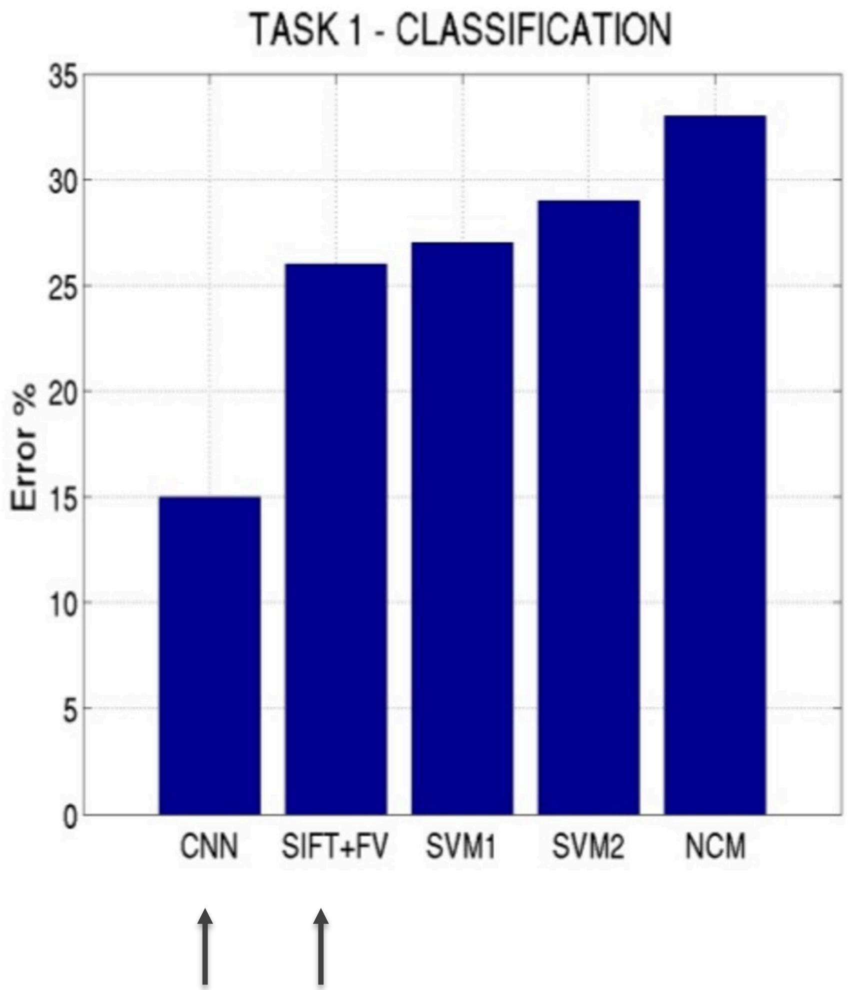
~1000 training images per Category

~1 million images in total for training

~50k images for validation

Only images released for the test set but no annotations, evaluation is performed centrally by the organizers

Top-5 error on this competition (2012 when things changed)



conv. neural network symbolic

Context (2011-2015 things changed)

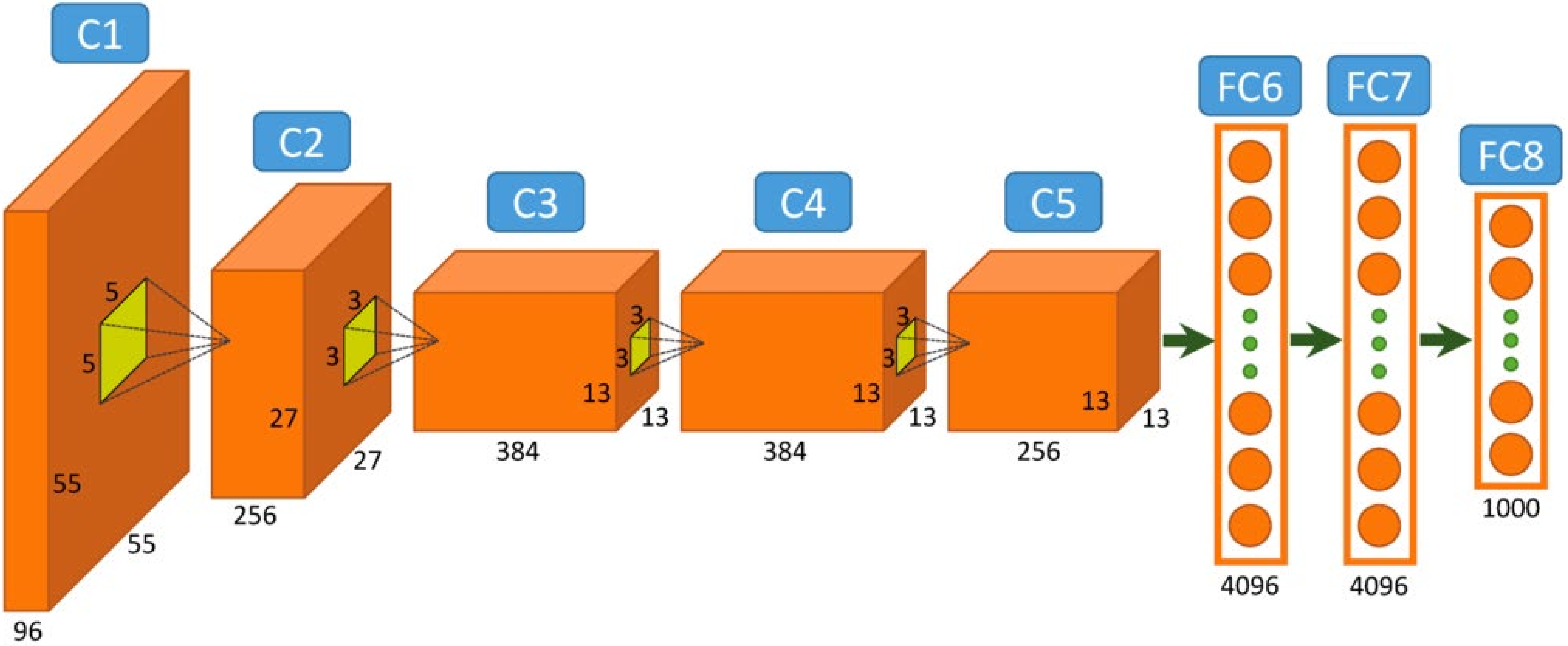
- Deep learning big successes

Team	Year	Place	Error (top-5)
XRCE (pre-neural-net explosion)	2011	1st	25.8%
Supervision (AlexNet)	2012	1st	16.4%
Clarifai	2013	1st	11.7%
GoogLeNet (Inception)	2014	1st	6.66%
Andrej Karpathy (human)	2014	N/A	5.1%
BN-Inception (Arxiv)	2015	N/A	4.9%
Inception-v3 (Arxiv)	2015	N/A	3.46%

Imagenet challenge classification task

From: http://www.wsdm-conference.org/2016/slides/WSDM2016-Jeff-Dean.pdfz

Alexnet



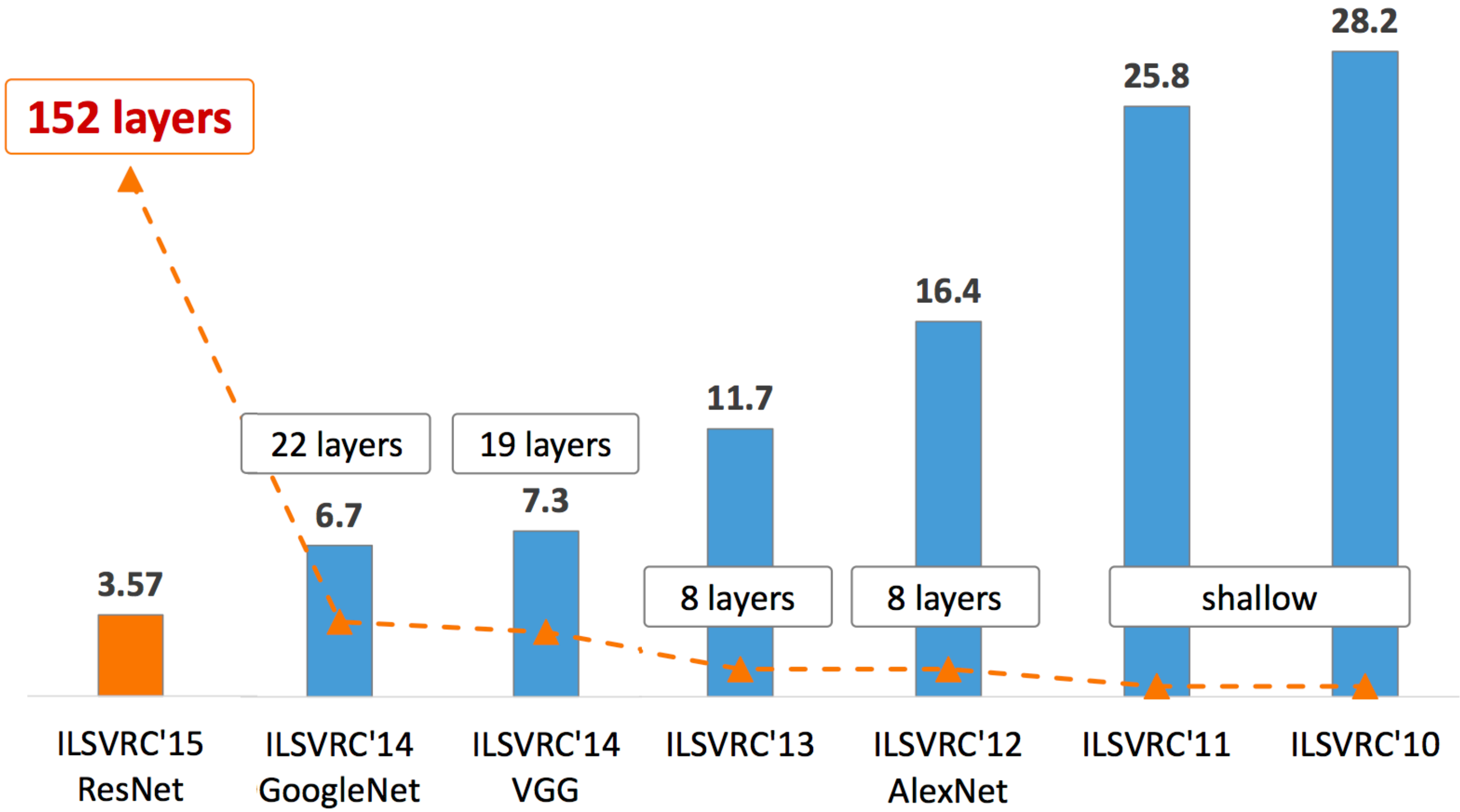
https://www.saagie.com/fr/blog/object-detection-part1

Revolution of Depth

AlexNet, 8 layers
(ILSVRC 2012)

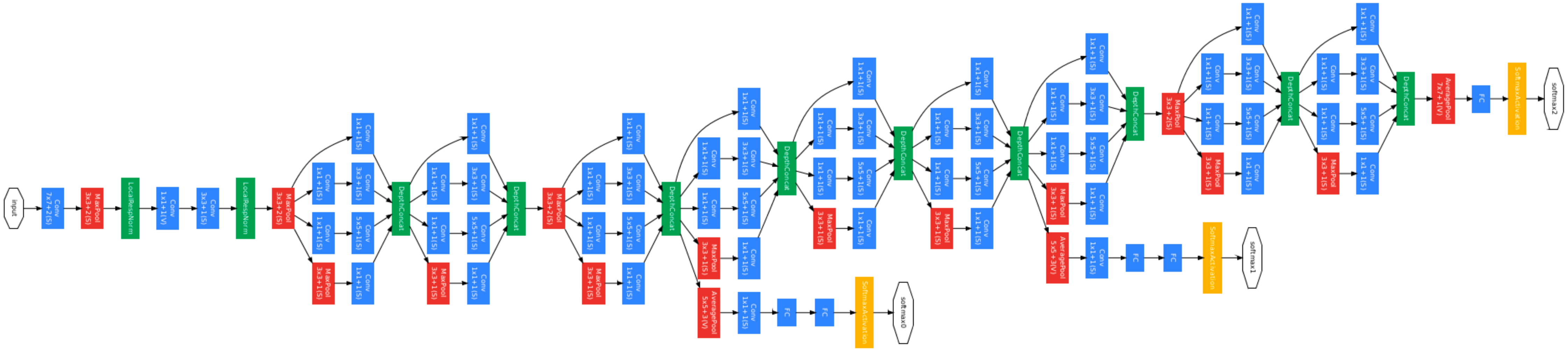
VGG, 19 layers
(ILSVRC 2014)

ResNet, 152 layers
(ILSVRC 2015)



UNIVERSITY OF CALGARY

GoogLeNet



Keras: https://gist.github.com/joelouismarino/a2ede9ab3928f999575423b9887abd14

Szegedy et al. 2014

UNIVERSITY OF CALGARY

ResNet

Sorry, does not fit in slide.

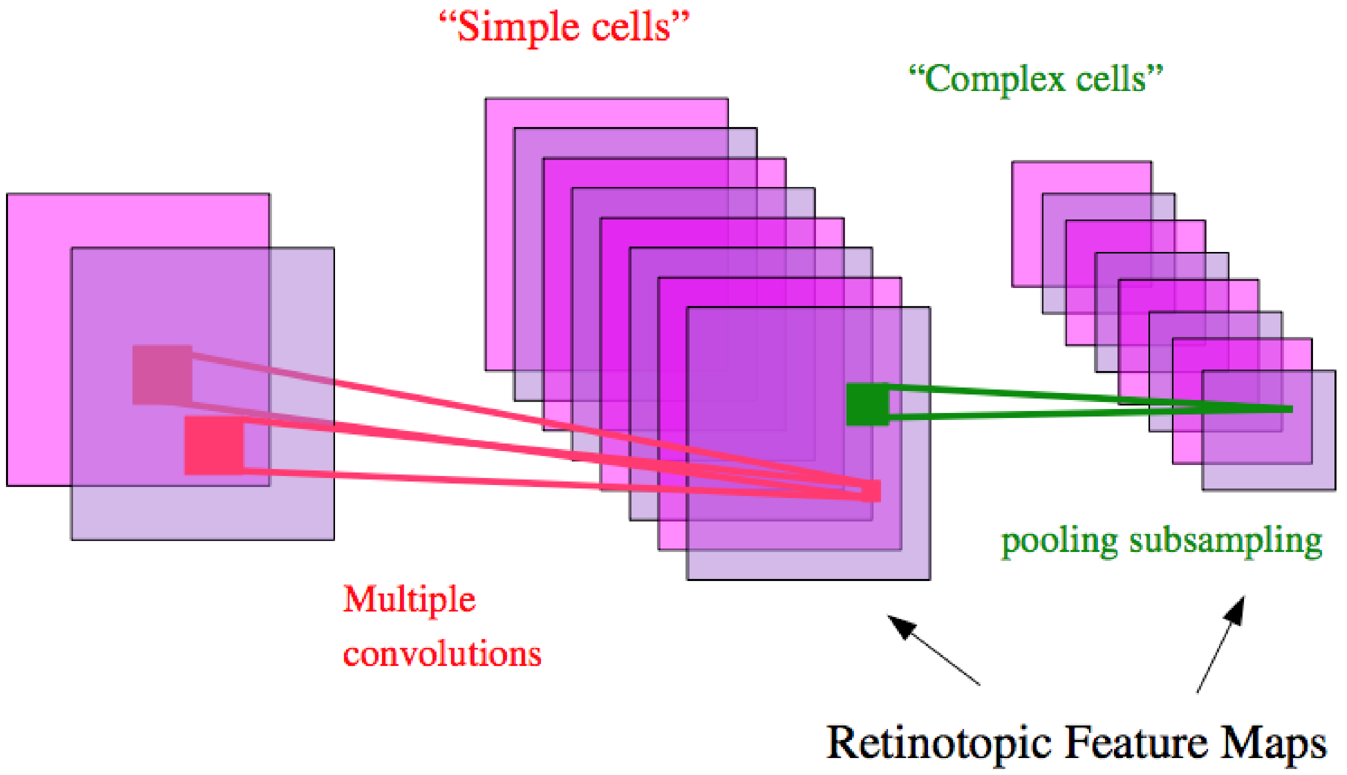
http://felixlaumon.github.io/assets/kaggle-right-whale/resnet.png

Keras: https://github.com/raghakot/keras-resnet/blob/master/resnet.py

Convolution

Model of vision in animals

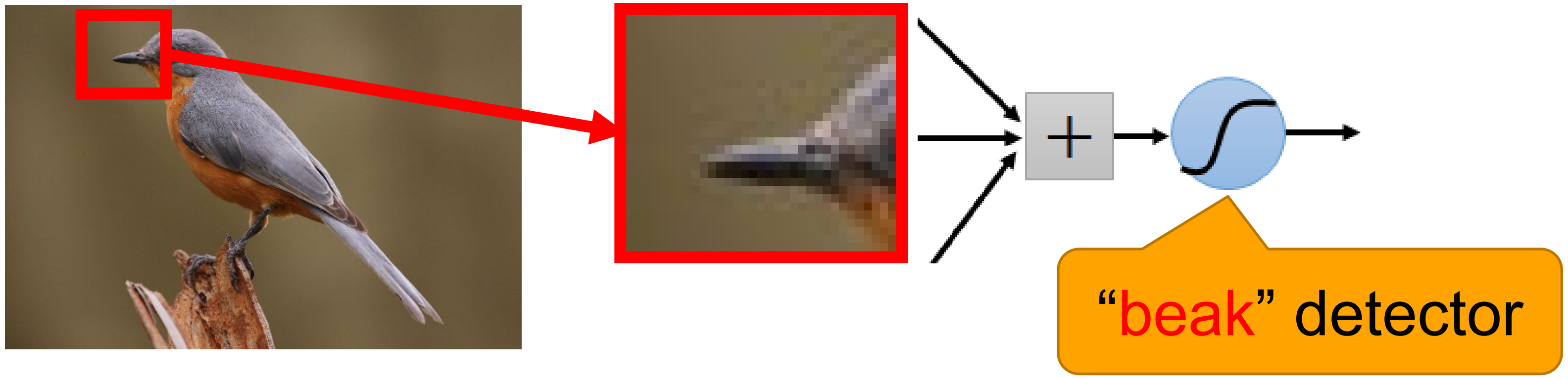
- [Hubel & Wiesel 1962]:
  - simple cells detect local features
  - complex cells "pool" the outputs of simple cells within a retinotopic neighborhood.



Consider learning an image:

- Some patterns are much smaller than the whole image

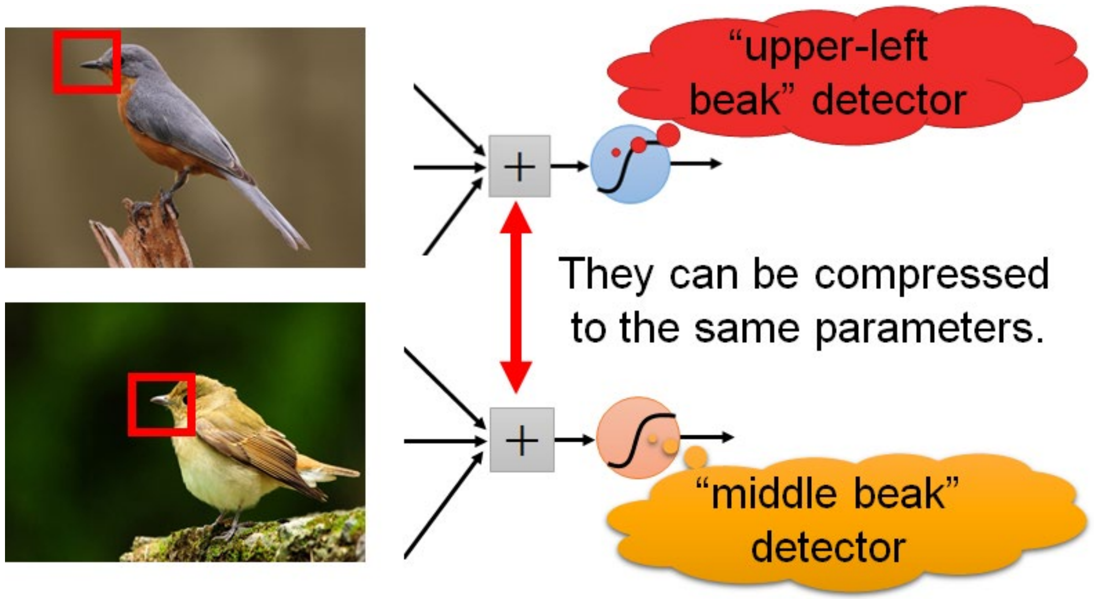
Can represent a small region with fewer parameters



UNIVERSITY OF CALGARY

Detectors

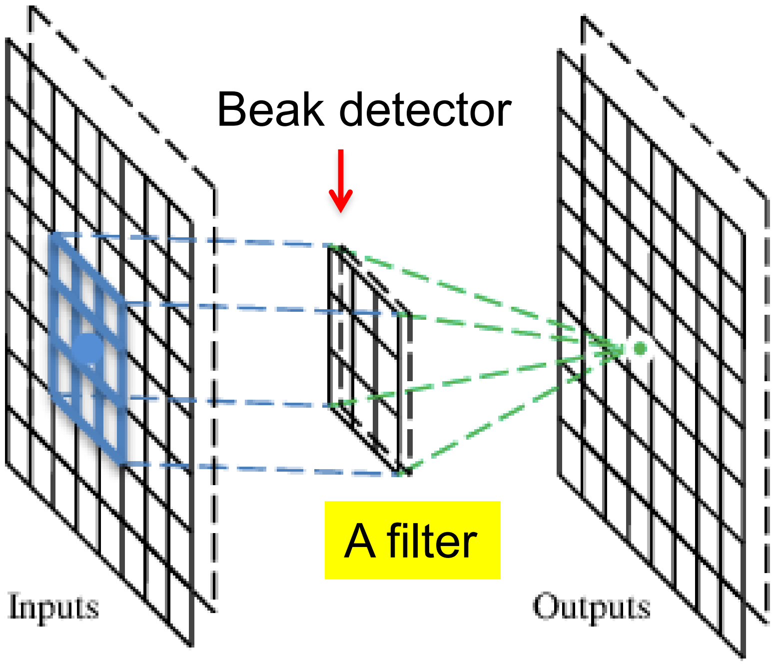
- Same pattern appears in different places:
  They can be compressed!
  What about training a lot of such "small" detectors
  and each detector must "move around".



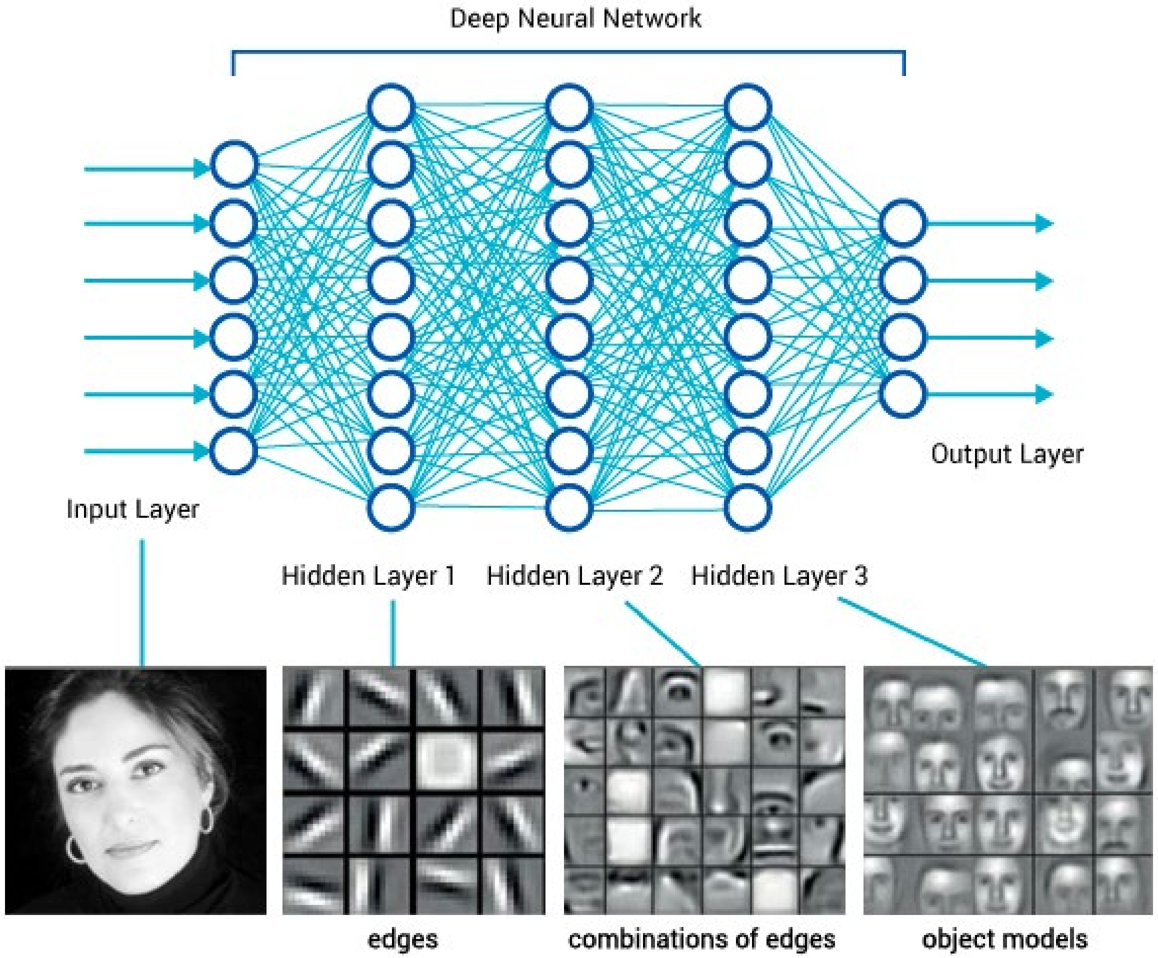
UNIVERSITY OF CALGARY

A convolutional layer

- A CNN is a neural network with some convolutional layers (and some other layers). A convolutional layer has a number of filters that does convolutional operation.



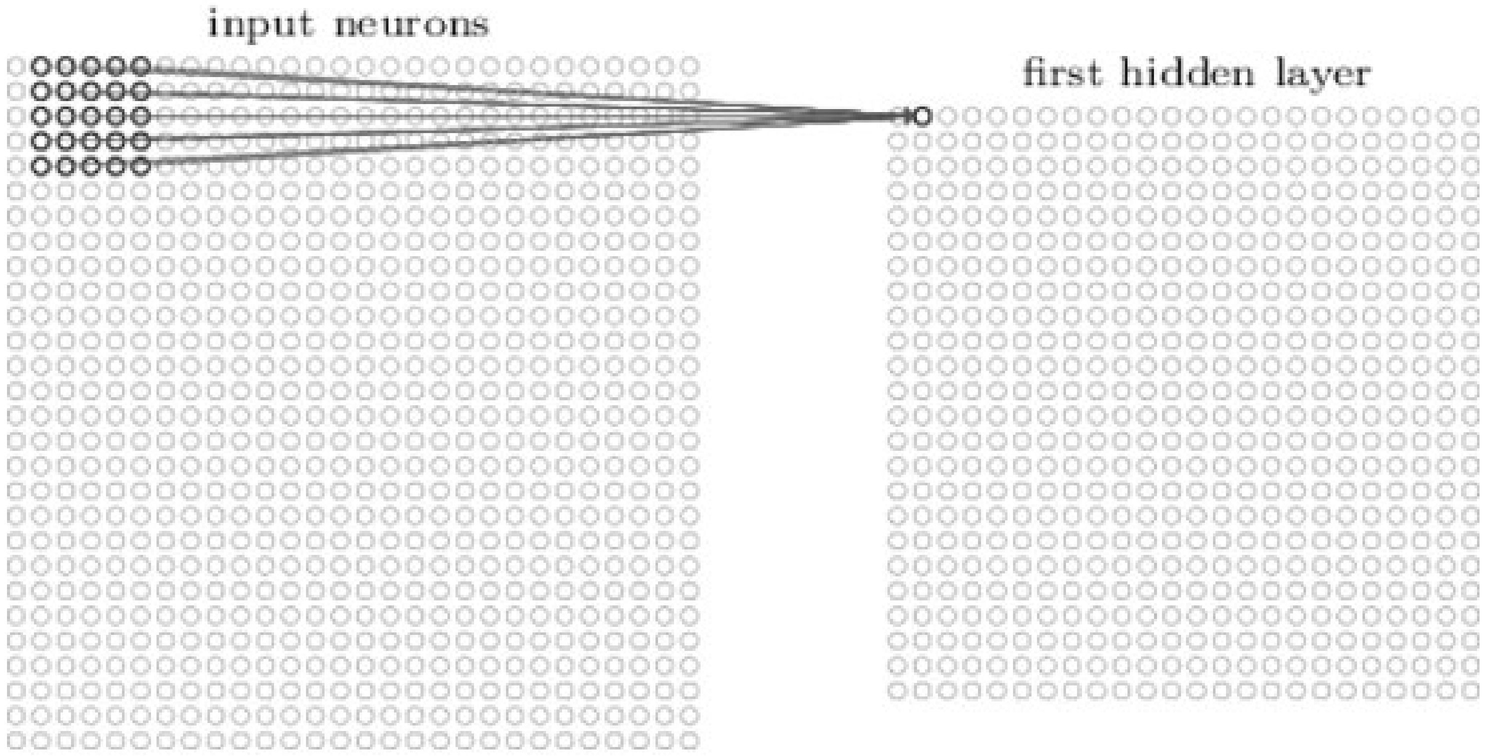
What is happening?



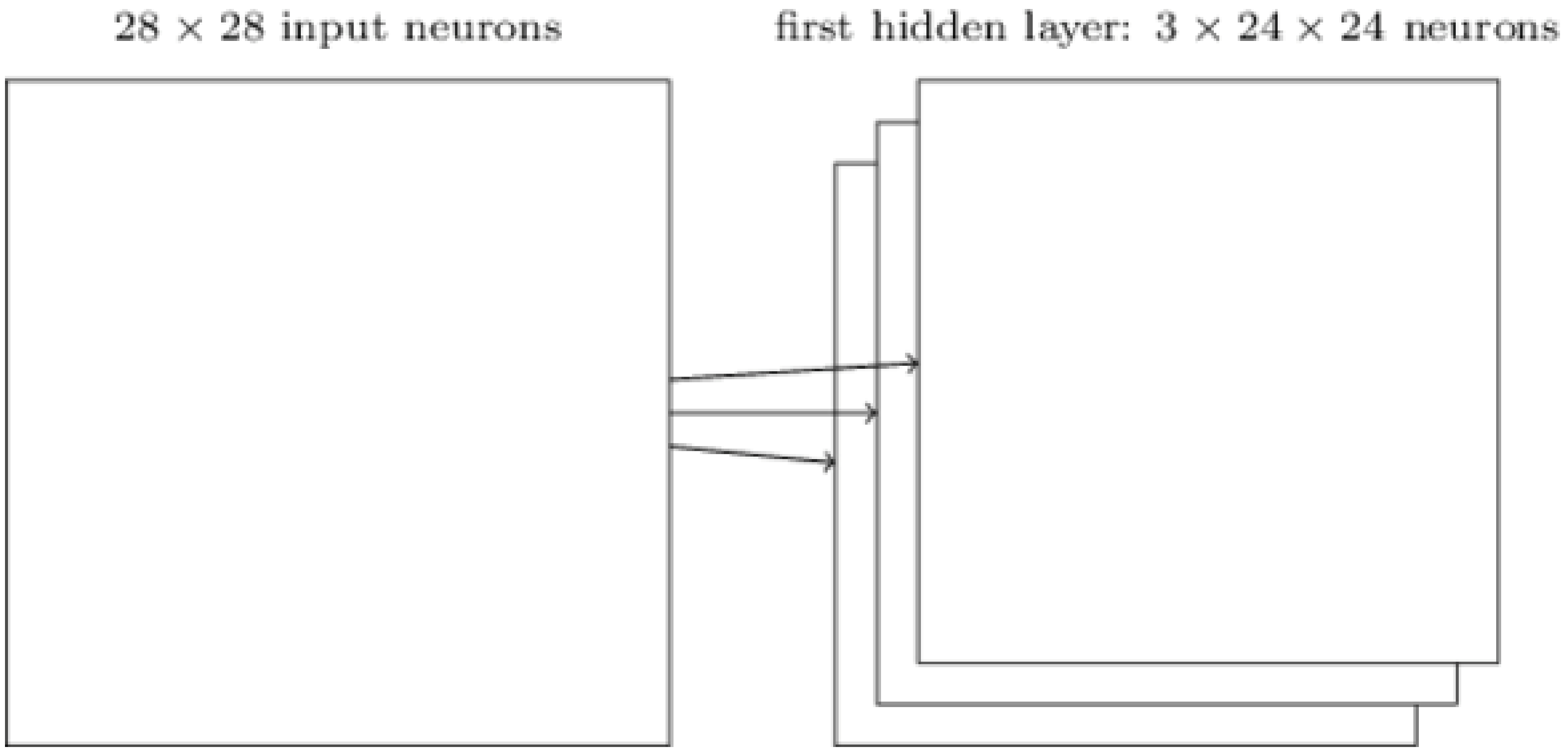
https://www.saagie.com/fr/blog/object-detection-part1

How do we convolve an image with an ANN?

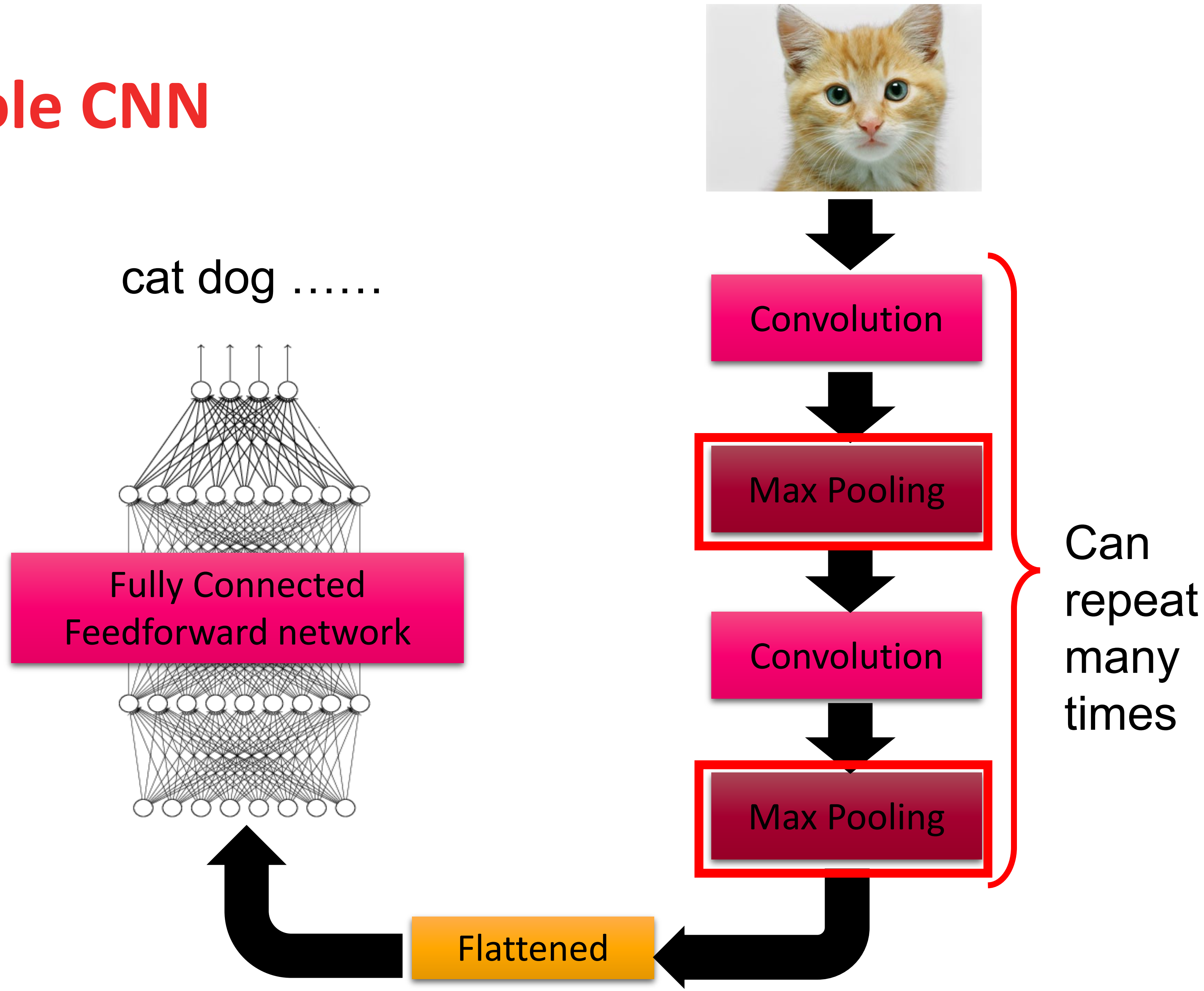
Note that the parameters in the matrix defining the convolution are **tied** across all places that it is used



How do we do many convolutions of an image with an ANN?



The whole CNN



Pooling

Why Pooling

- Subsampling pixels will not change the object

bird

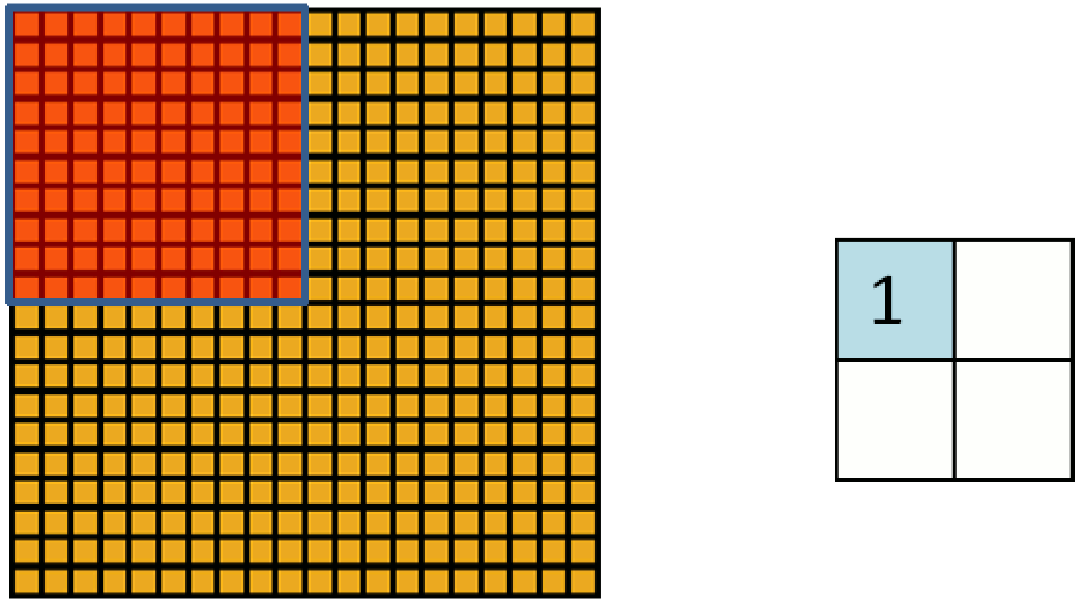
Subsampling

bird

We can subsample the pixels to make image smaller

→ fewer parameters to characterize the image

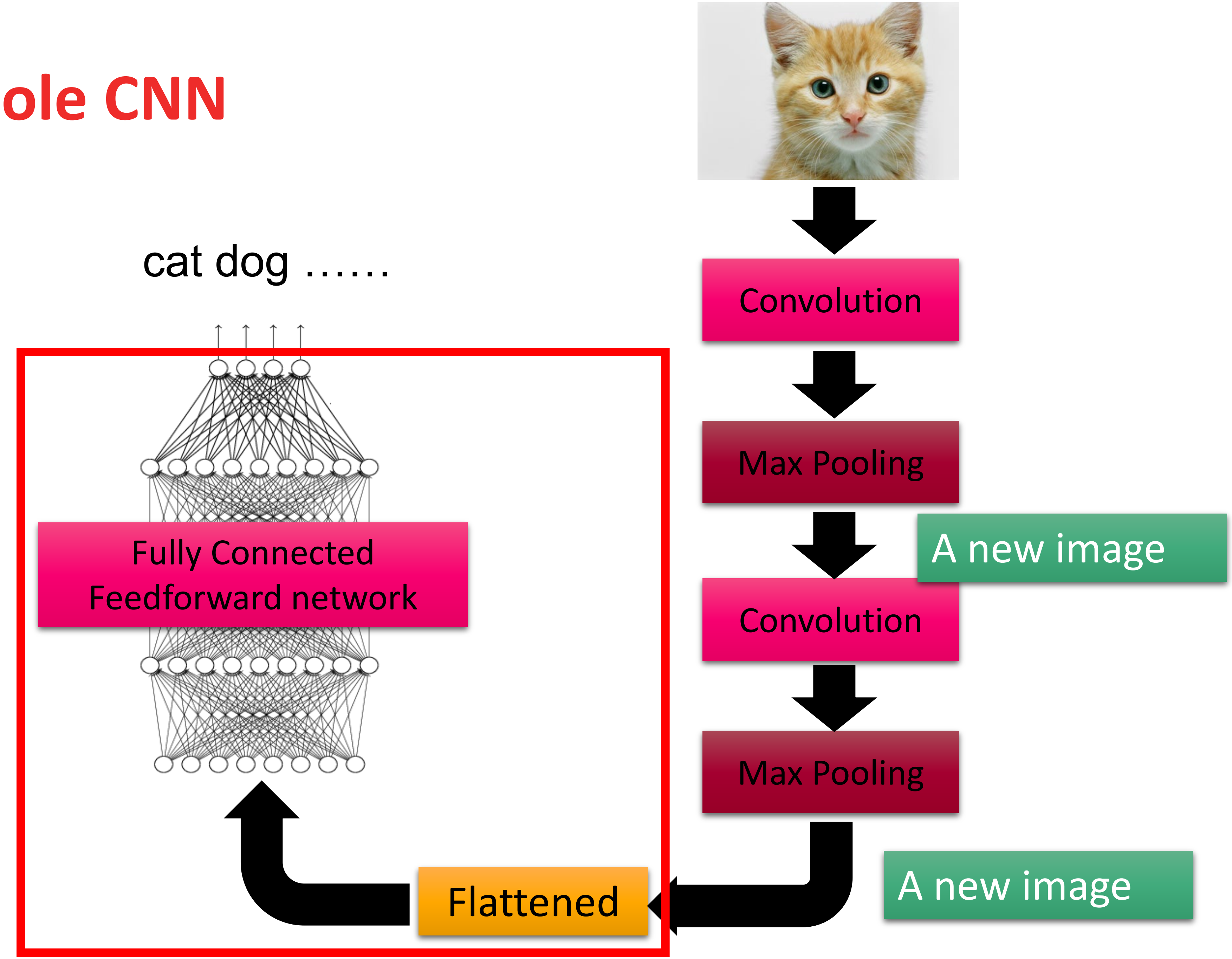
Pooling



Convolved feature

Pooled feature

The whole CNN



Examples CNN

AlphaGo

19 x 19 matrix

Black: 1

white: -1

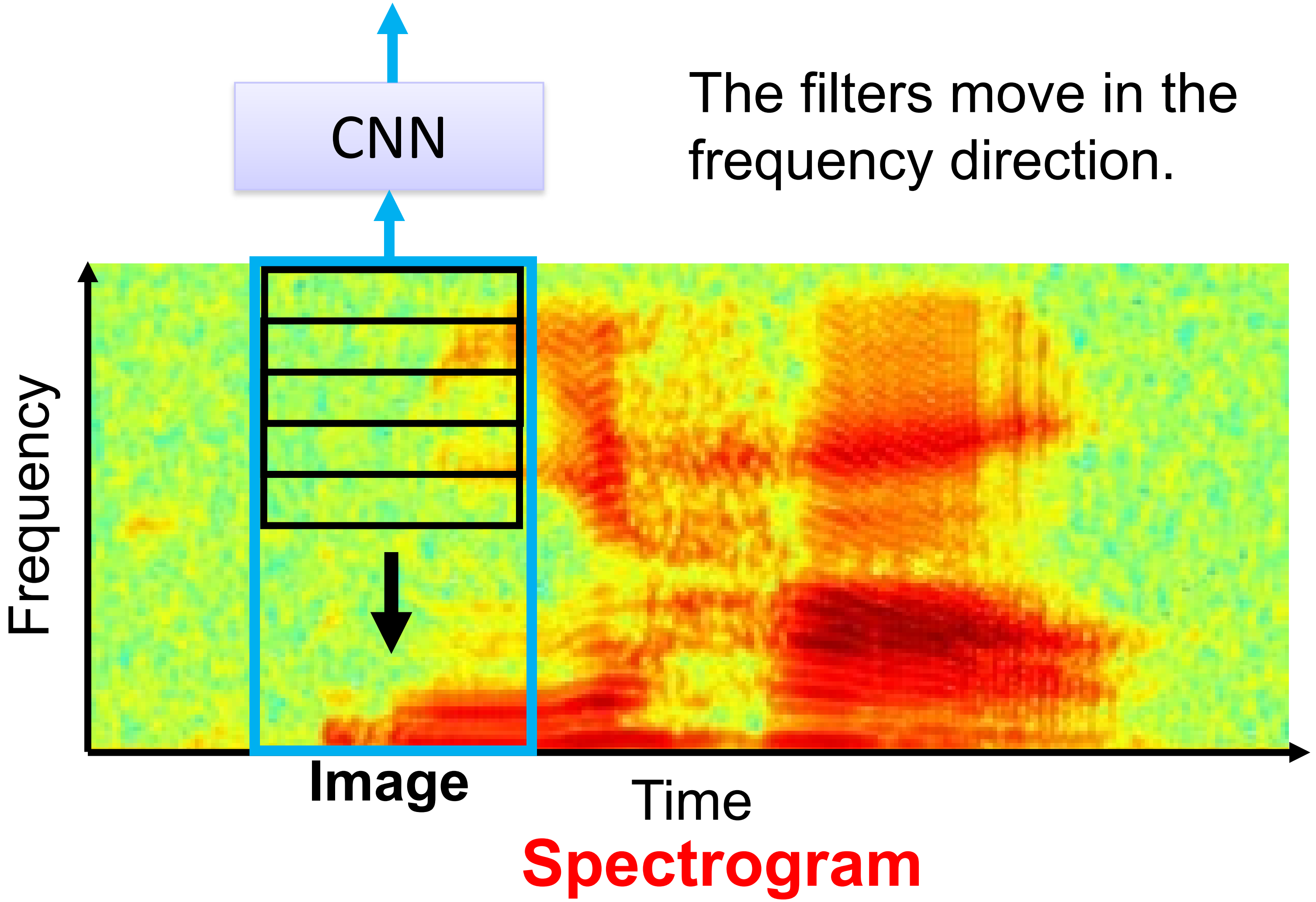
none: 0

→ Neural Network → Next move (19 x 19 positions)

Fully-connected feedforward network can be used

But CNN performs much better

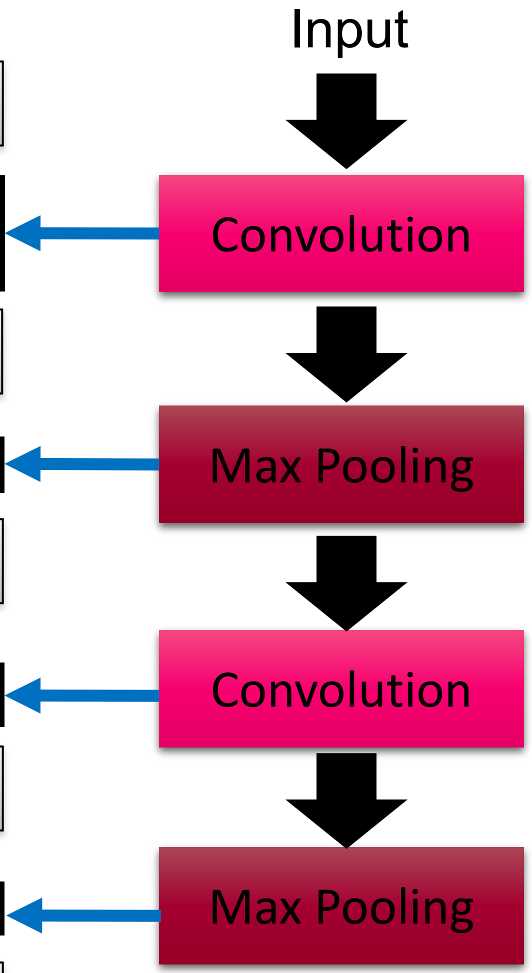
CNN in speech recognition



The filters move in the frequency direction.

CNN in Keras

Only modified the ***network structure*** and ***input format (vector -> 3-D array)***



1 x 28 x 28

```
model2.add( Convolution2D( 25,3,3,
            input_shape=(28,28,1)) )
```

25 x 26 x 26

How many parameters for each filter?

```
model2.add(MaxPooling2D((2,2)))
```

25 x 13 x 13

```
model2.add(Convolution2D(50,3,3))
```

50 x 11 x 11

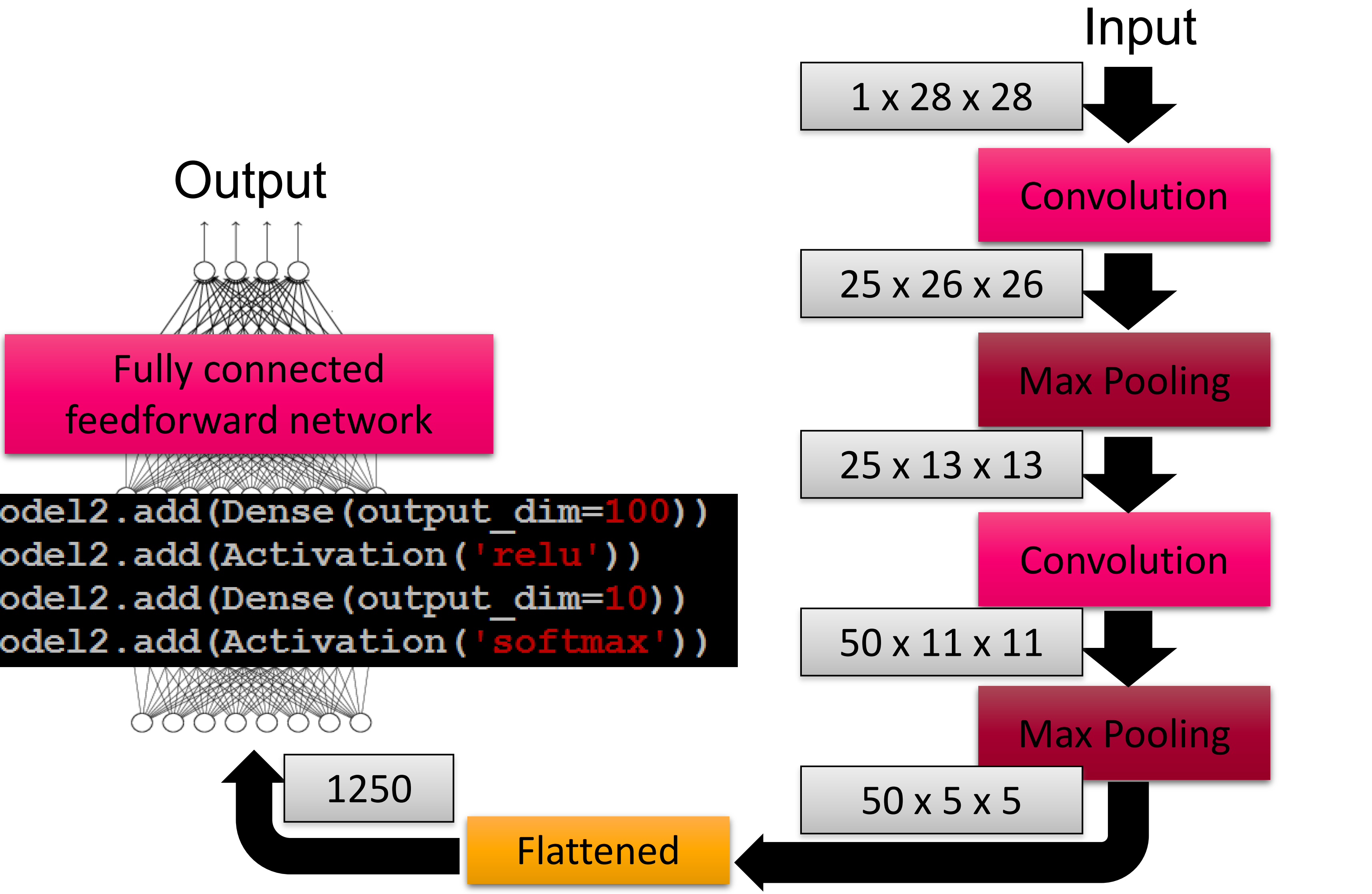
How many parameters for each filter?

```
model2.add(MaxPooling2D((2,2)))
```

50 x 5 x 5

CNN in Keras

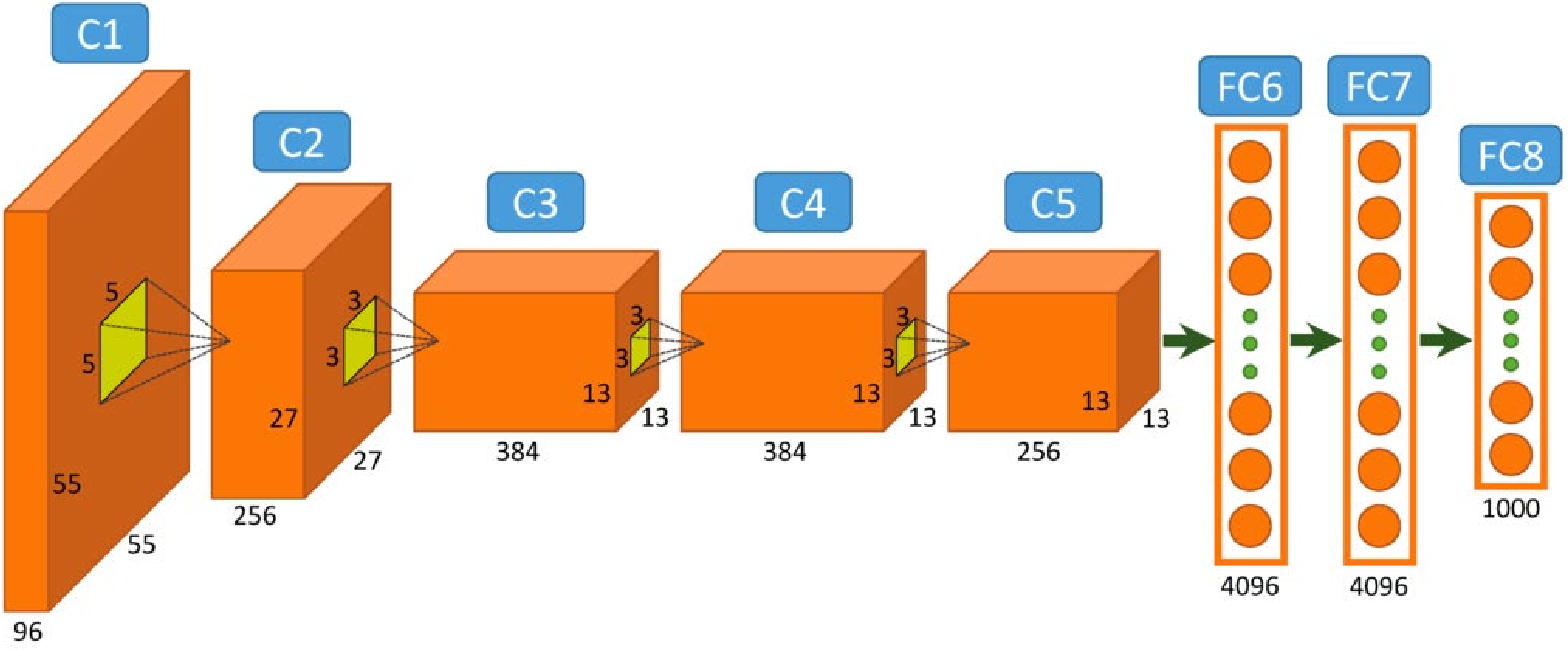
Only modified the ***network structure*** and ***input format (vector -> 3-D array)***




```
model2.add(Dense(output_dim=100))
model2.add(Activation('relu'))
model2.add(Dense(output_dim=10))
model2.add(Activation('softmax'))
```

```
model2.add(Flatten())
```

Alexnet



https://www.saagie.com/fr/blog/object-detection-part1

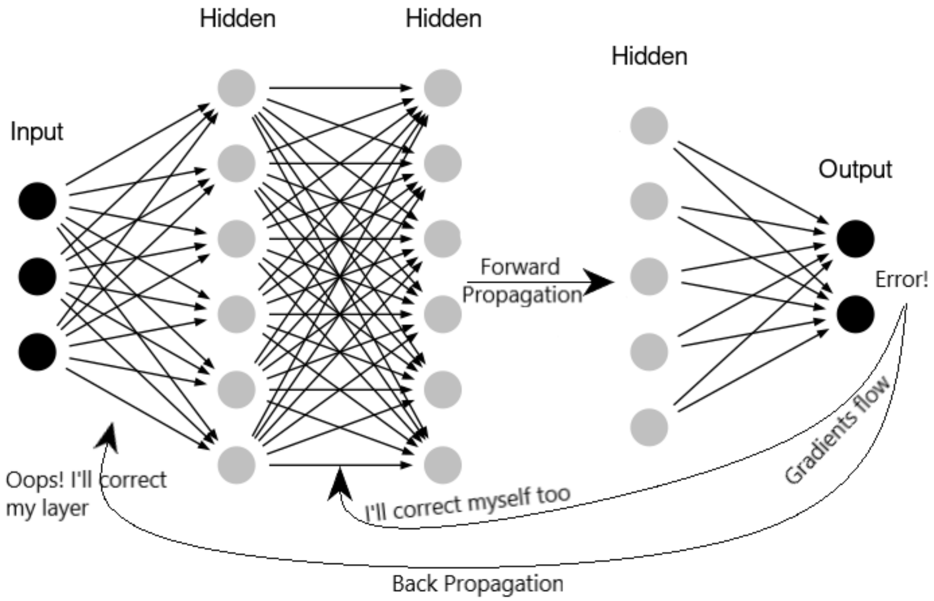
Alexnet in Keras

```
model = Sequential()
model.add(Convolution2D(64, 3, 11, 11, border_mode='full'))
model.add(BatchNormalization((64,226,226)))
model.add(Activation('relu'))
model.add(MaxPooling2D(poolsize=(3, 3)))

model.add(Convolution2D(128, 64, 7, 7, border_mode='full'))
model.add(BatchNormalization((128,115,115)))
model.add(Activation('relu'))
model.add(MaxPooling2D(poolsize=(3, 3)))

model.add(Convolution2D(192, 128, 3, 3, border_mode='full'))
model.add(BatchNormalization((128,112,112)))
model.add(Activation('relu'))
model.add(MaxPooling2D(poolsize=(3, 3)))

model.add(Convolution2D(256, 192, 3, 3, border_mode='full'))
model.add(BatchNormalization((128,108,108)))
model.add(Activation('relu'))
model.add(MaxPooling2D(poolsize=(3, 3)))

model.add(Flatten())
model.add(Dense(12*12*256, 4096, init='normal'))
model.add(BatchNormalization(4096))
model.add(Activation('relu'))
model.add(Dense(4096, 4096, init='normal'))
model.add(BatchNormalization(4096))
model.add(Activation('relu'))
model.add(Dense(4096, 1000, init='normal'))
model.add(BatchNormalization(1000))
model.add(Activation('softmax'))
```

Batch Normalization

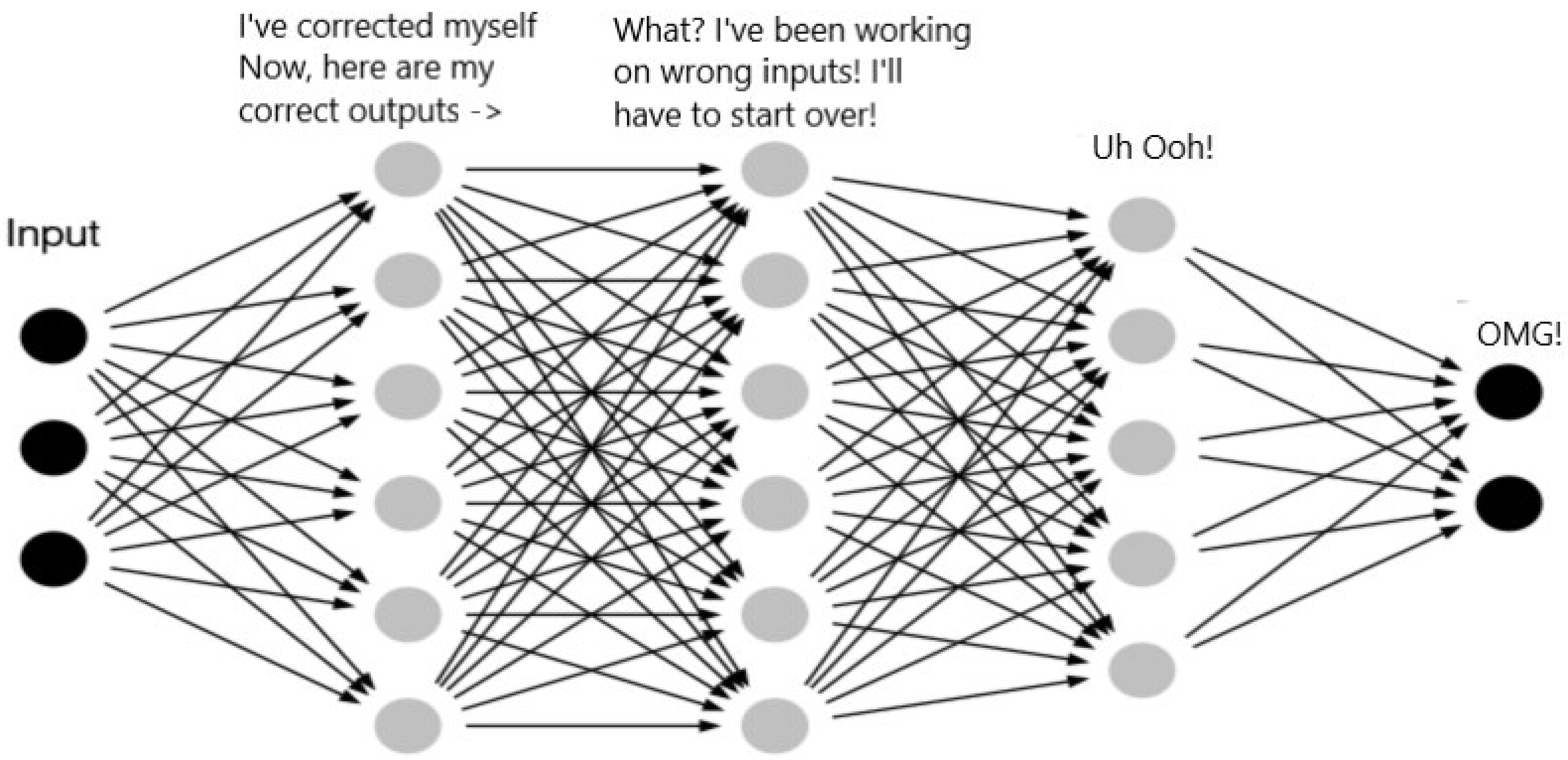
Batch Normalization?

- Neural networks learn the problem using BackPropagation algorithm.
- BackPropagation involves computing gradients for each layer
- In deep networks this time explodes for training

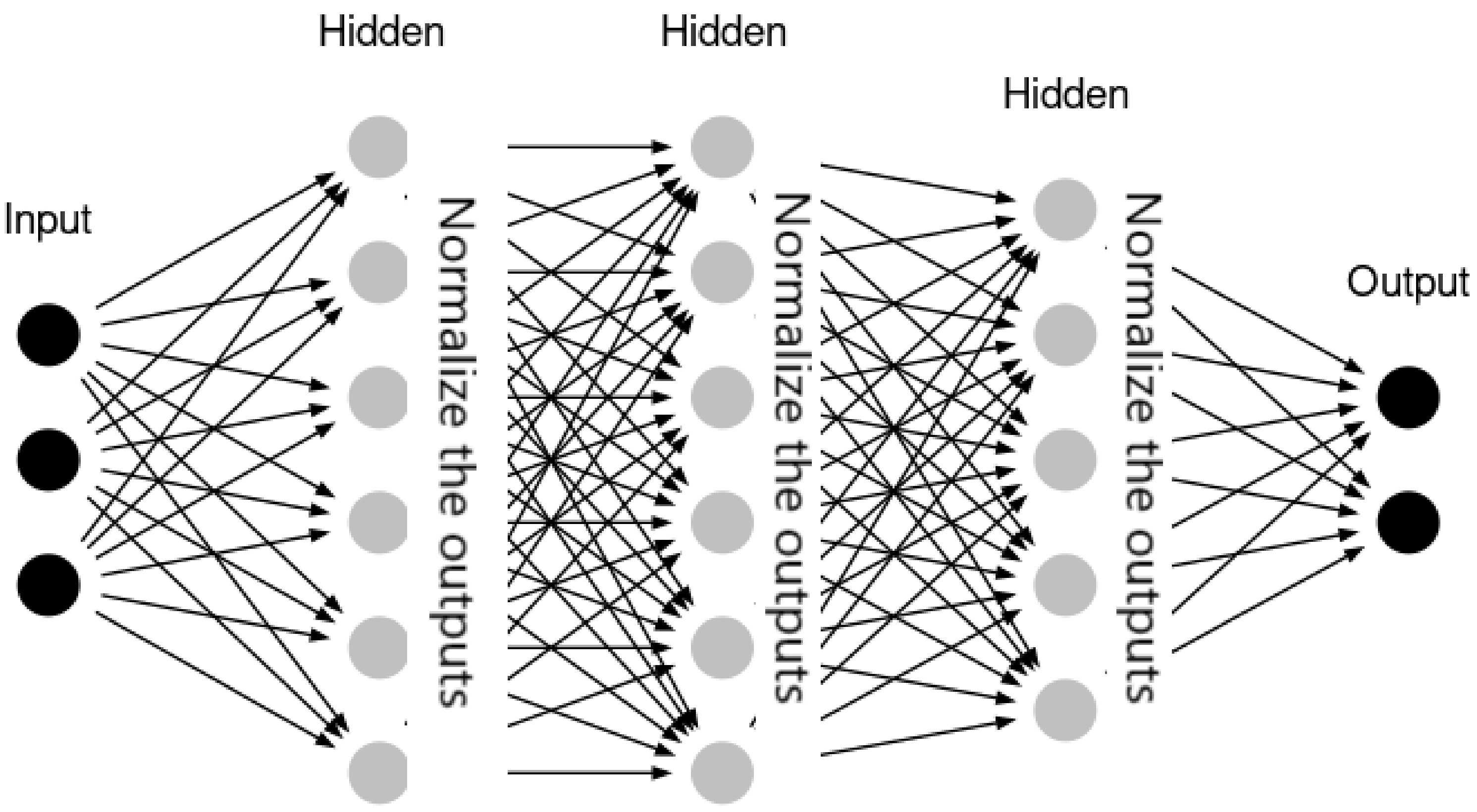


UNIVERSITY OF CALGARY

Batch Normalization?



Batch Normalization?



35

UNIVERSITY OF CALGARY

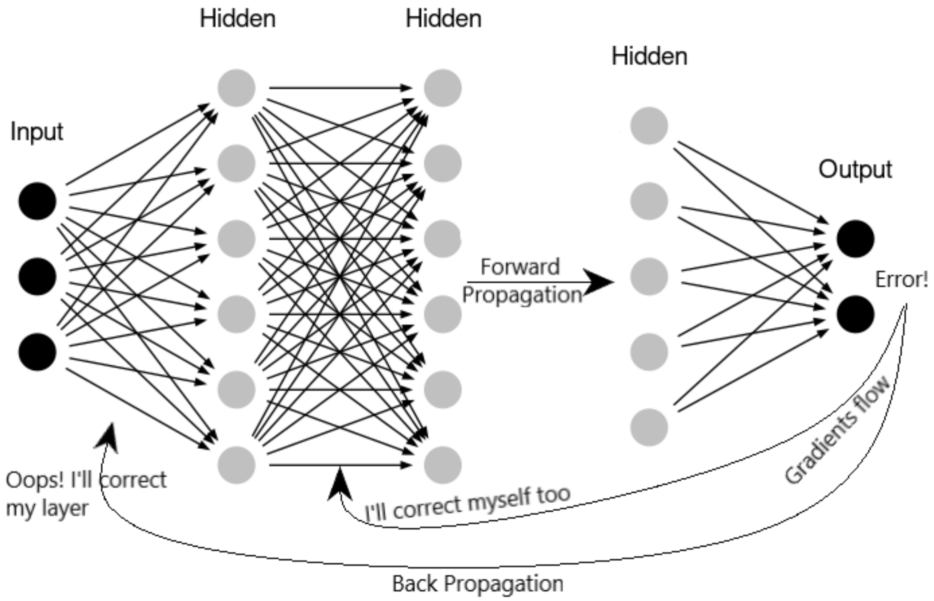
Batch Normalization?

- Normalization brings all the inputs centered around 0.
- This way, there is not much change in each layer input.
- So, layers in the network can learn from the back-propagation simultaneously, without waiting for the previous layer to learn.
- This speeds up the training of networks.

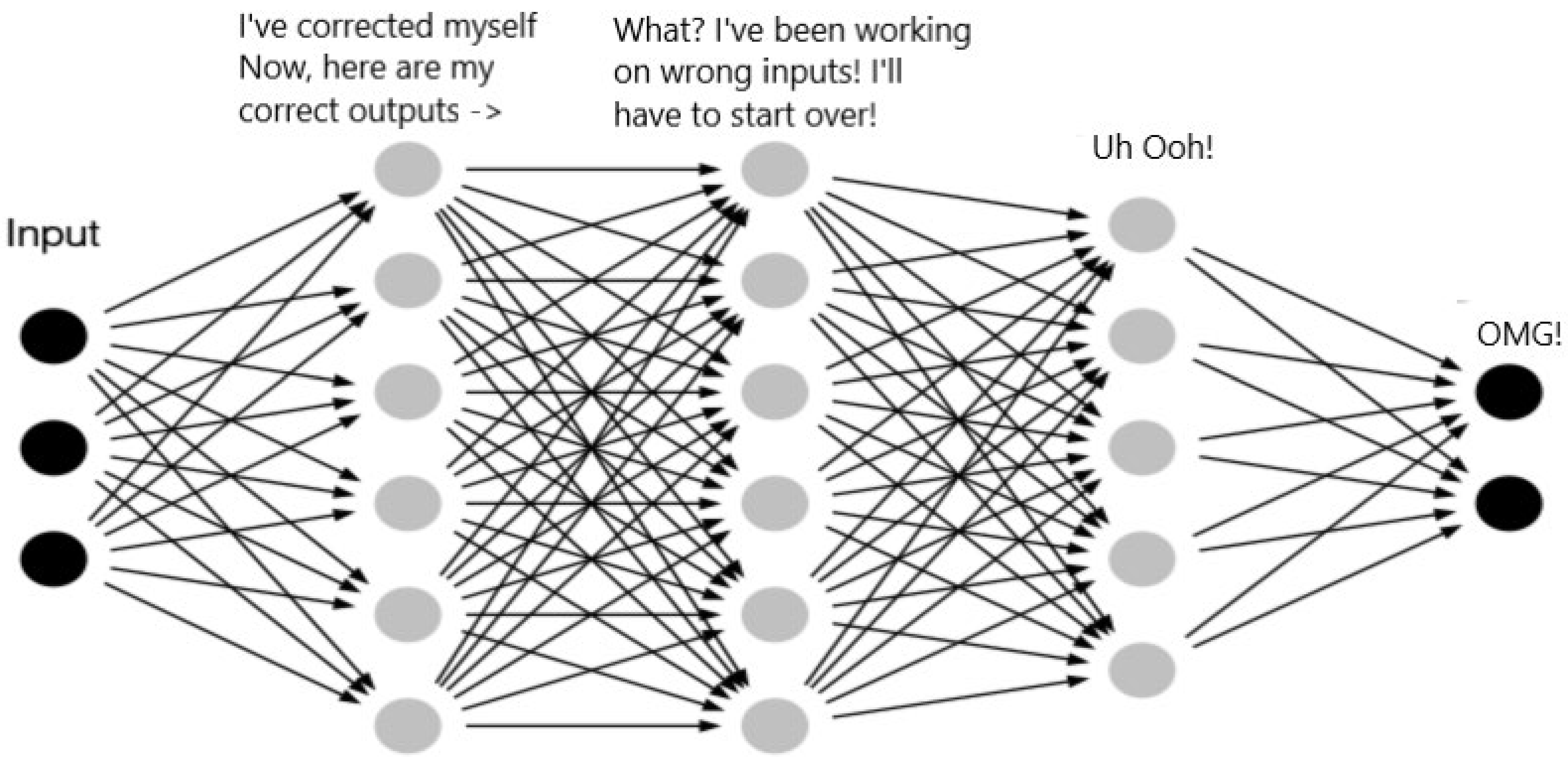
36

Batch Normalization?

- Neural networks learn the problem using BackPropagation algorithm.
- BackPropagation involves computing gradients for each layer
- In deep networks this time explodes for training

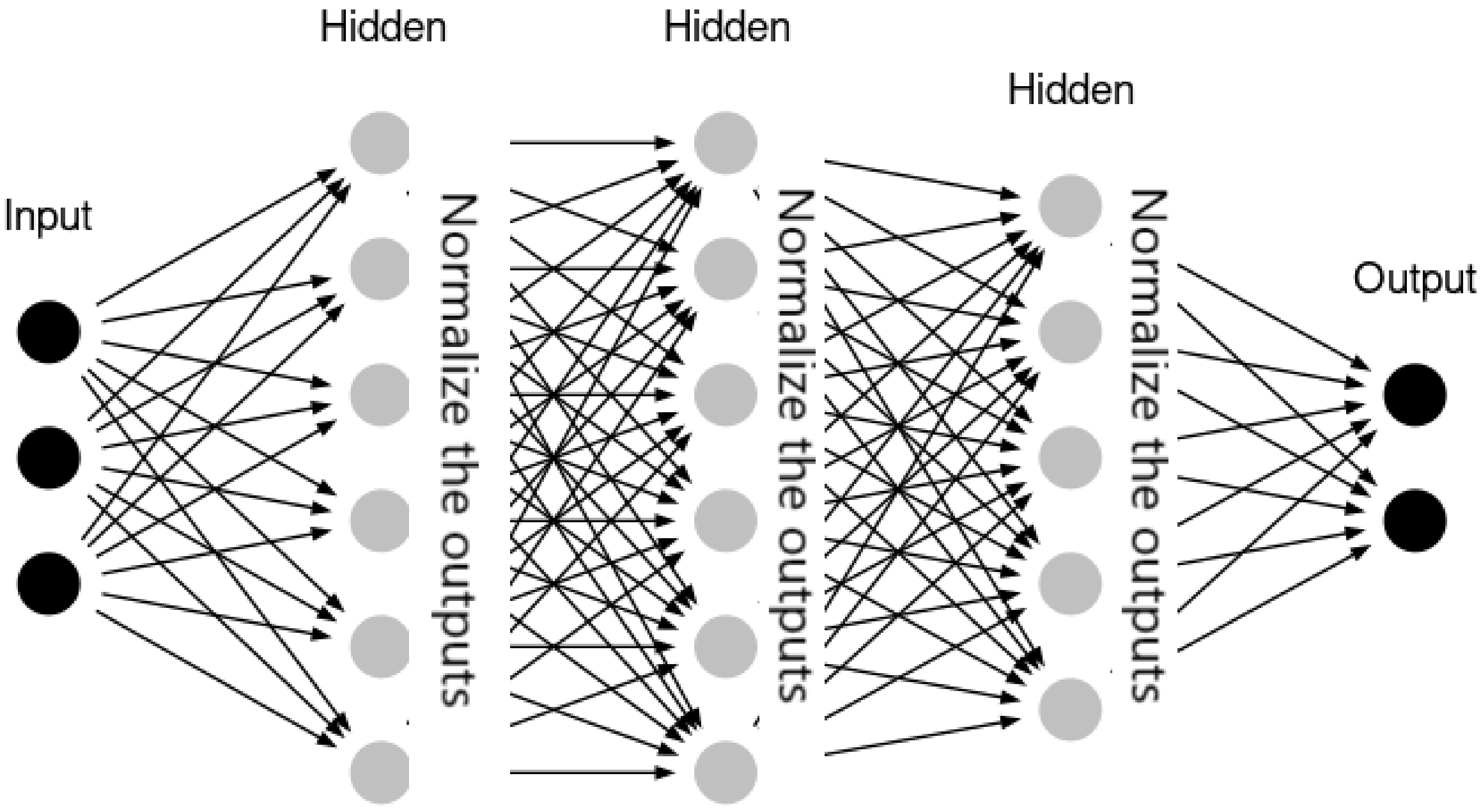


Batch Normalization?



UNIVERSITY OF CALGARY

Batch Normalization?



Batch Normalization?

- Normalization brings all the inputs centered around 0.
- This way, there is not much change in each layer input.
- So, layers in the network can learn from the back-propagation simultaneously, without waiting for the previous layer to learn.
- This speeds up the training of networks.

Pre-Processing

Preprocessing and Data Augmentation

Preprocessing and Data Augmentation

256

256

Preprocessing and Data Augmentation

224x224

UNIVERSITY OF CALGARY

Preprocessing and Data Augmentation

224x224

UNIVERSITY OF CALGARY





True label: Abyssinian cat

Reflection on ImageNet

ImageNet

- Not designed for people
- Recently went viral
- Sept 23, 2019
- "ImageNet will remove 600,000 images of people stored on its database after an art project exposed racial bias in the program's artificial intelligence system."

ImageNet

- First presented as a research poster in 2009
- Scraped a collection of many millions of images from the internet
- Trained through images categorized by Amazon Mechanical Turk workers
- Crowdsourcing platform through which people can earn money performing small tasks
- Sorted an average of 50 images per minute into thousands of categories
- In 2012, a team from the University of Toronto used a Convolutional Neural Network to handily win the top prize
- Final year 2017, and accuracy in classifying objects in the limited subset had risen from 71.8% to 97.3%. (That did not include "Person" category)

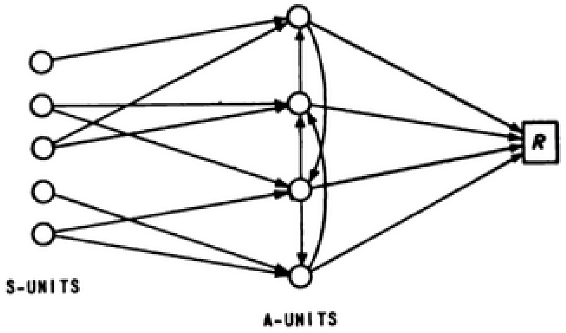
ImageNet

- AI researcher Kate Crawford and artist Trevor Paglen
  - Training Humans — an exhibition that at the Prada Foundation in Milan
  - Part of their experiment also lives online at ImageNet Roulette, a website where users can upload their own photographs to see how the database might categorize them.
  - https://www.excavating.ai/
- Example of the complexities and dangers of human classification
- The sliding spectrum between supposedly unproblematic labels like “trumpeter” or “tennis player” to concepts like “spastic,” “mulatto,” or “redneck.”
- ImageNet is an object lesson in what happens when people are categorized like objects.
  - Not all ‘nouns’ are equal

Other Network Types

Recurrent Neural Networks

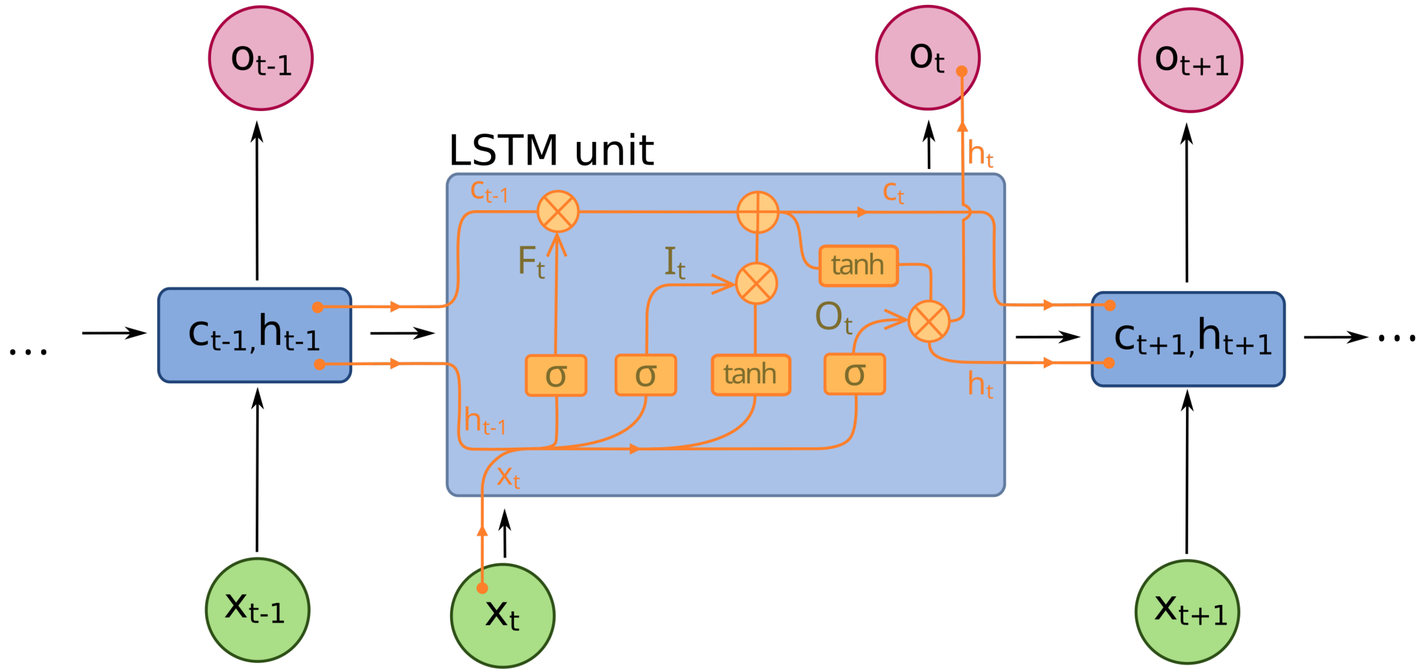
- 1901"recurrent semicircles" observed in the cerebellar cortex
- 1940s, multiple people proposed the existence of feedback in the brain, in contrast to previous presumptions of feedforward structure.
- The McCulloch and Pitts paper (1943), which proposed the McCulloch-Pitts neuron model, considered networks that contains cycles
- 1960s Rosenblatt 3-layered network with cycles



- Modern RNN networks are mainly LSTM (1995)
  - and BRNN (bi-directional).
- 2006, BRNN started to revolutionize speech recognition
- Early 2010s encoder-decoder sequence (two RNN) to do machine translation
  - Led to development of Transformers

LTSM

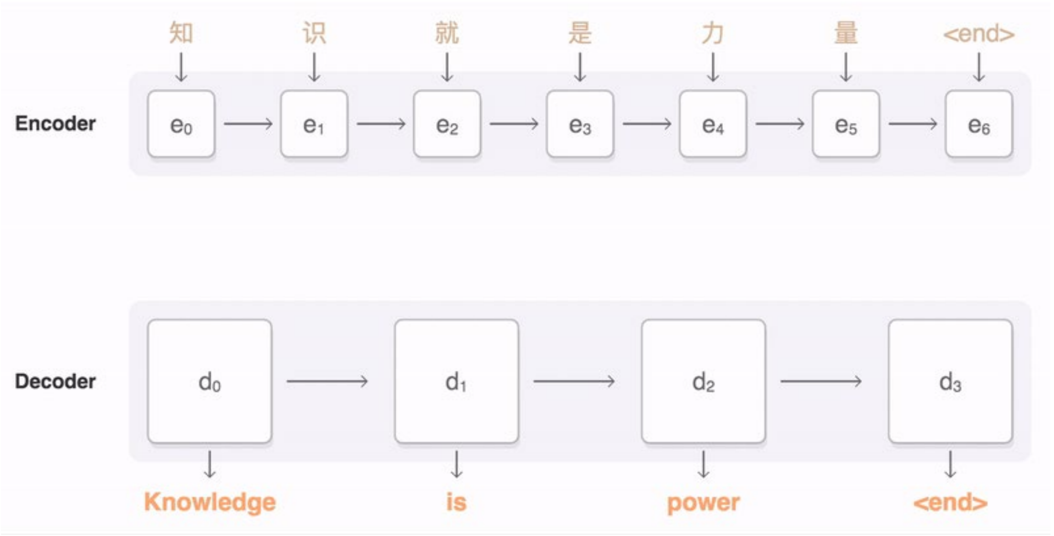
- Long short-term memory (LSTM) is the most widely used RNN architecture
- The unit acts as a storage buffer
- Ex. Instead of a NN that predicts one point unique each time for one input
    - You can feed a sequence of inputs through it, and the prior points will also be used to influence the ones that follow
    - Good for prediction of things with history
        - Words, paths, data signals

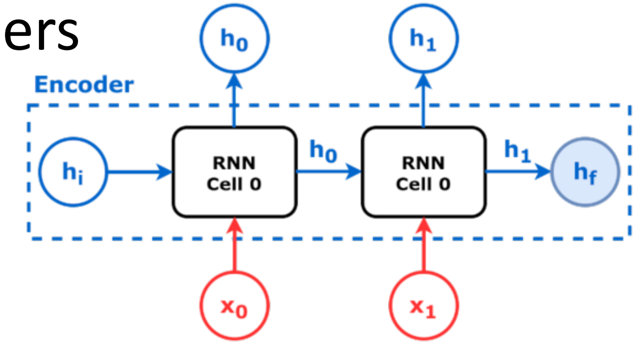


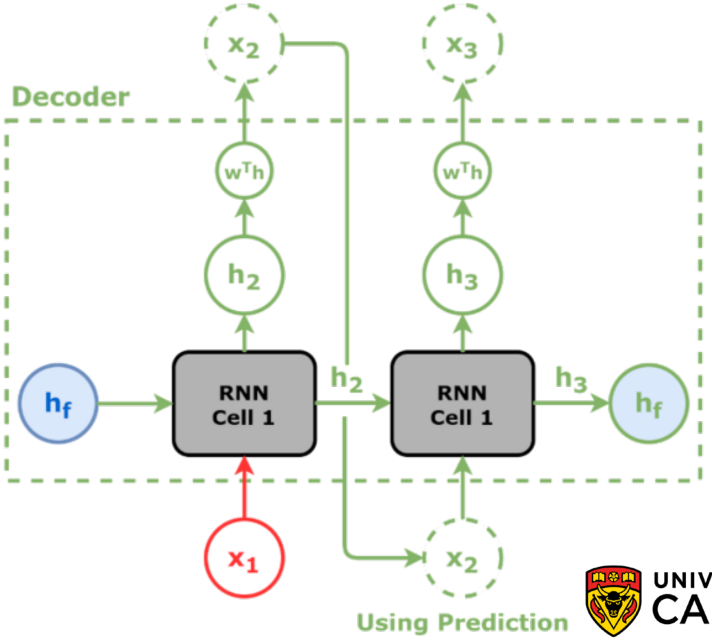
UNIVERSITY OF CALGARY

Encoder-Decoder

- Seq2seq problems require one sequence to be changed into another 2014
- In NN, encoder captures input to **latent vector state**
- Decoder takes a **latent vector** and generates output
  - retains consideration of prior symbols in sequence
- Issues with long input
- Standard until 2017 Transformers

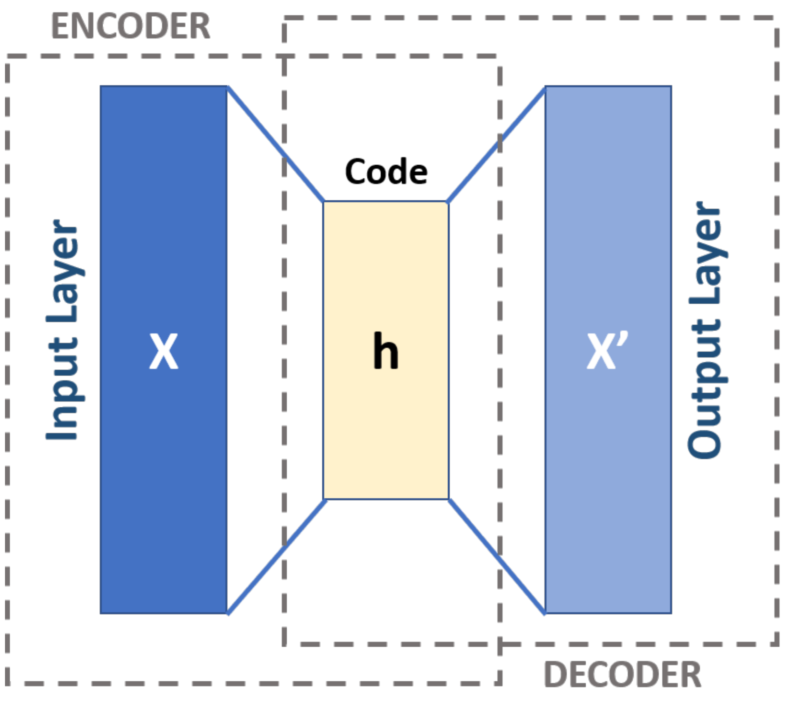






Auto-encoders

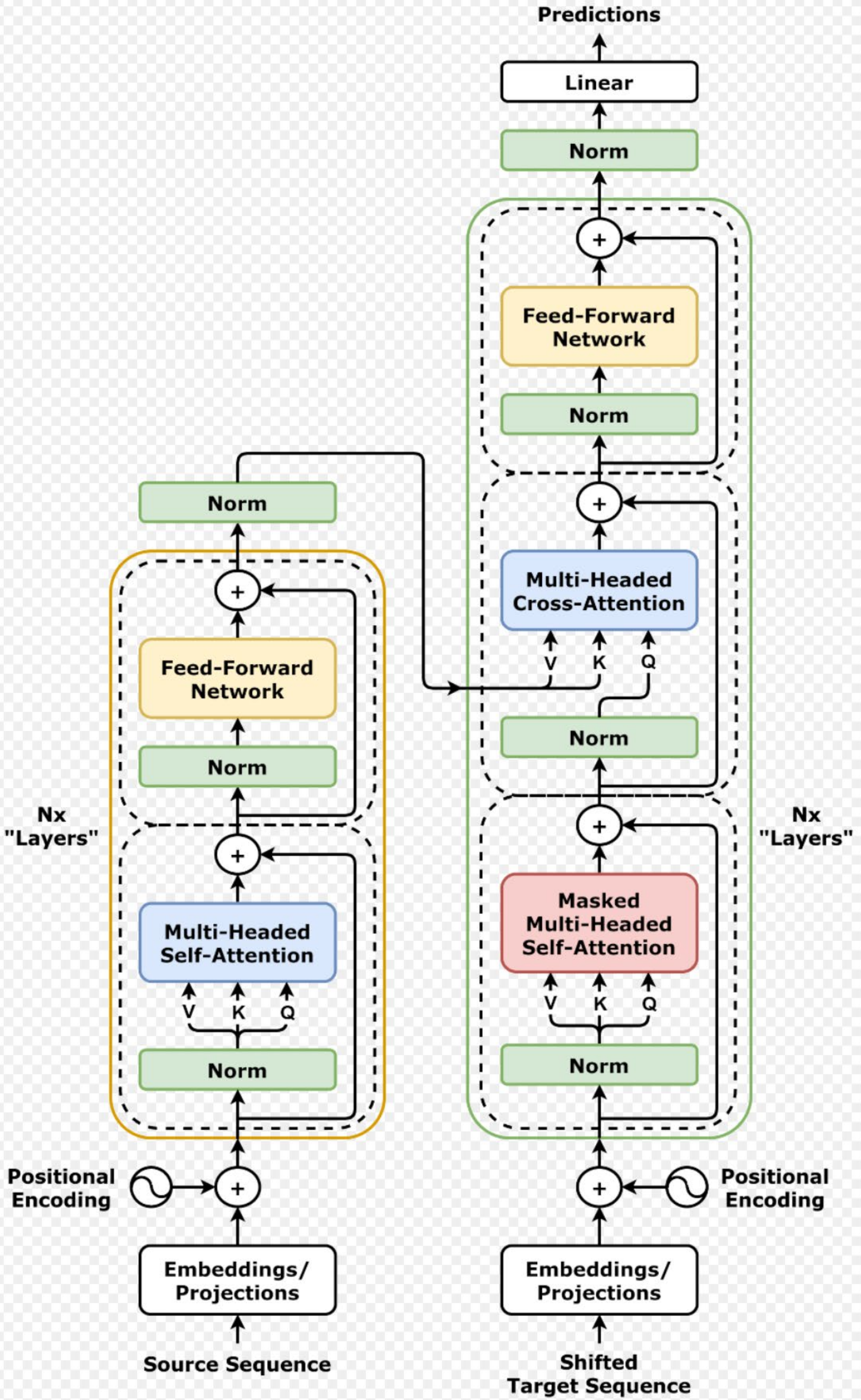
- When input/output the same, subset of encoder-decoder
- An autoencoder learns two functions:
  1. an encoding function that transforms the input data
  2. a decoding function that recreates the input data from the encoded representation.
- Can be used for facial recognition, feature detection, anomaly detection, and learning the meaning of words
- Randomly generate new data that is similar to the input (training) data
  - Even to do compression images, audio
  - https://arstechnica.com/information-technology/2022/09/better-than-jpeg-researcher-discovers-that-stable-diffusion-can-compress-images/



Transformers

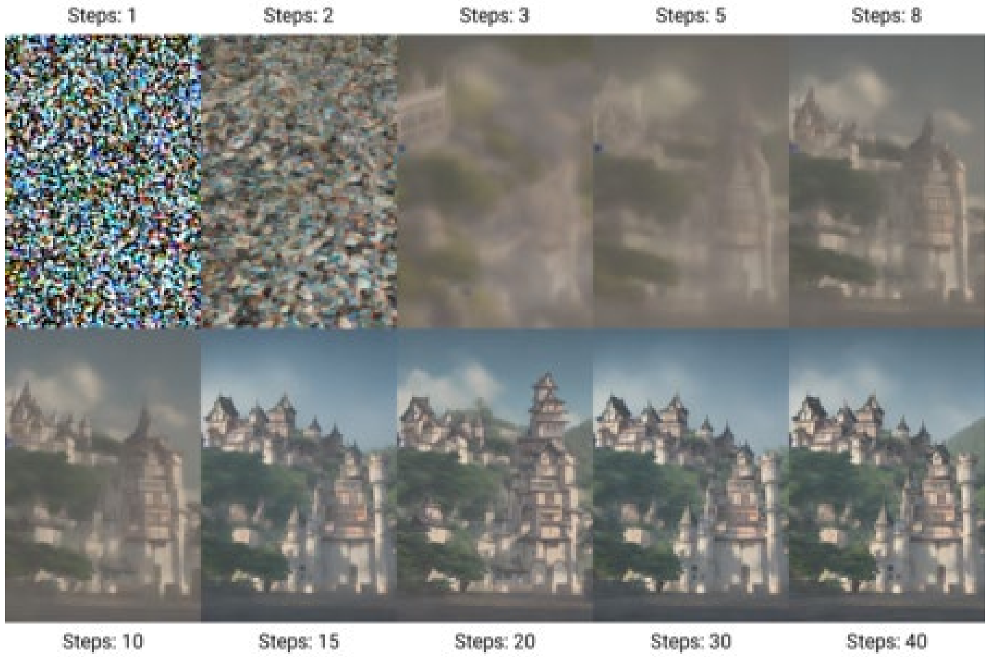
- Encoder-decoder transformer model (2017)
- Instead of one-sequence at a time can operate in parallel on sequence
  - This comes with quadratic scale in costs

- 2018 OpenAI GPT (Generative Pre-trained Transformer) for NLP generation
- 2019 google using it on search queries
- 2020 google translate using it
- 2020 GPT-3 visibility booms LLMs
- (also usable with images)
  - DALL-E (2021), Stable Diffusion and others



Latent Diffusion Model

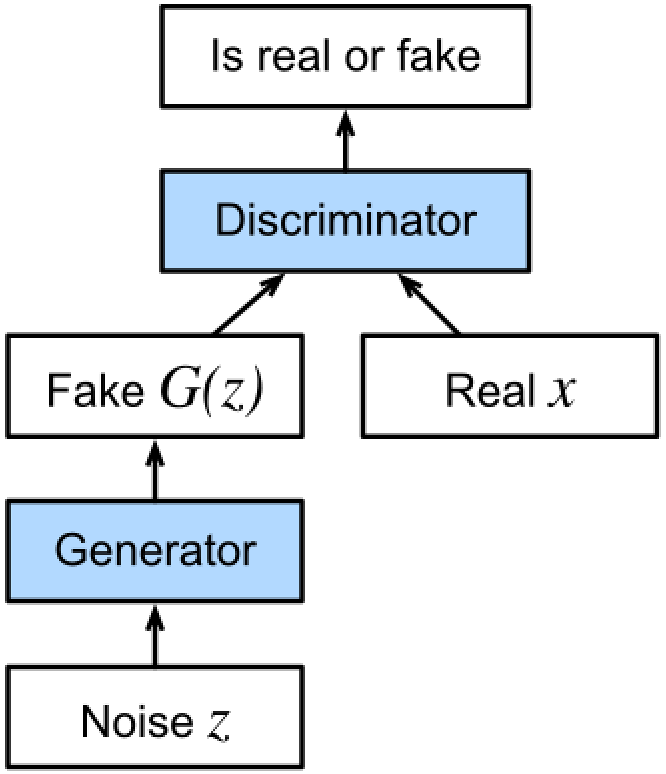
- Latent Diffusion Model (2015)
- Trained by gradually adding noise to the training images. The model is then trained to reverse this process, starting with a noisy image and gradually removing the noise until it recovers the original image
  - Like a sequence of denoising autoencoders
- The resulting embeds within itself a 'latent' or neural network concept of an image (this can be connected to text input as well for prompting)
- And LDM is used by asking for the idea of a latent output, the LDM then runs on some noisy input until the output is 'denoised' sufficiently



Generative Adversarial Networks

- Generative adversarial network (GAN)
- two neural networks contest with each other in the form of a zero-sum game,
  - where one agent's gain is another agent's loss.
  - Discriminator (labeler), Generator (maker)

- GANs are similar to mimicry in evolutionary biology, with an evolutionary arms race between both networks.



Discussion

Discussion

- Decentralized knowledge representation
  ☞ possibility to parallelize (GPUs!)
- Can find pattern outside of human understanding
- Currently best way to deal with sensory data

– Network structure determines what can be learned
  ☞ must be provided by user

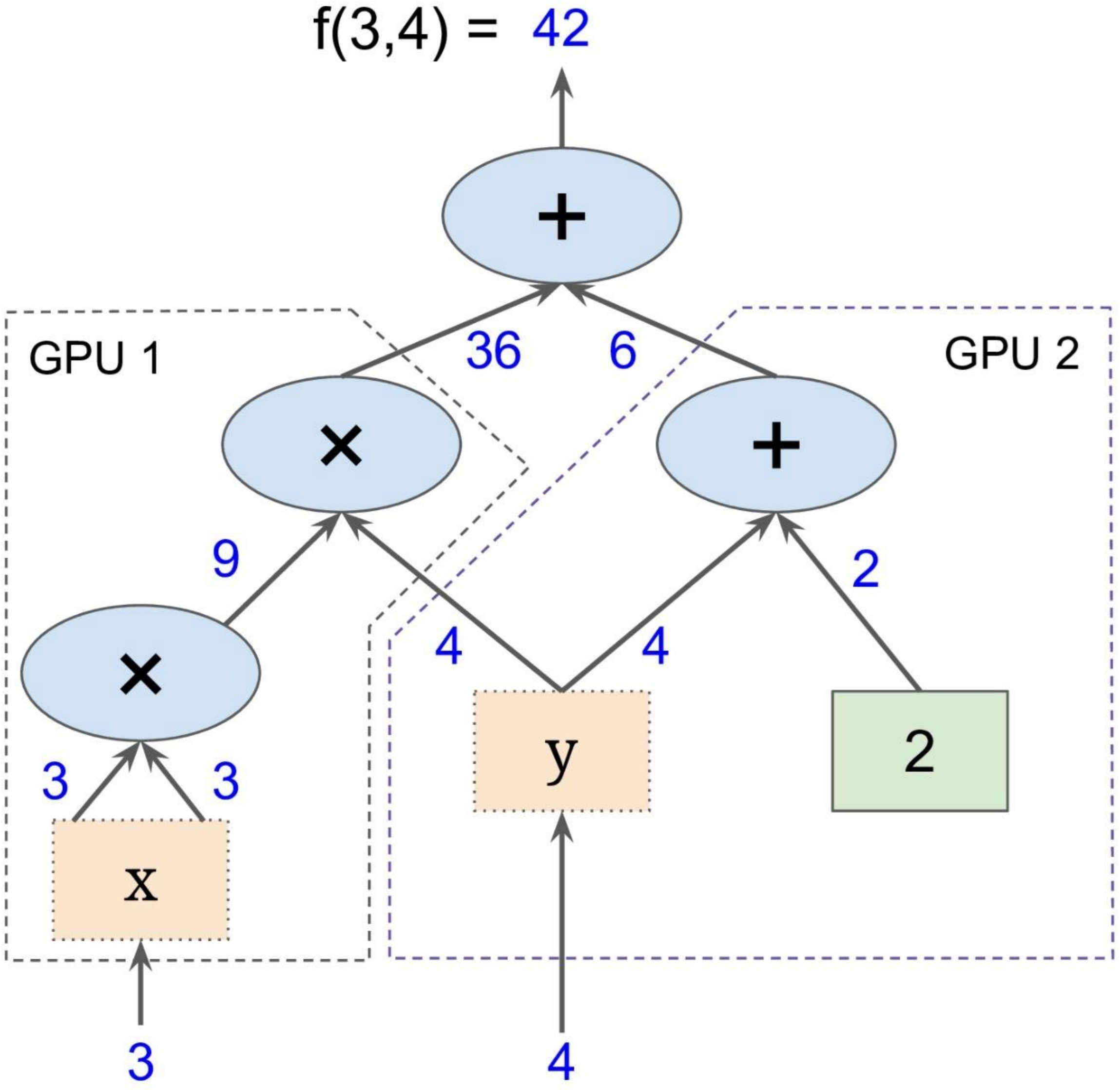
– Represented knowledge not understandable by humans

– Learning can take very long

– Too many learning procedures: when to choose which?

NN Bonus! -> Subgraphs Let us use Compute Units

Possible to break graphs into several chunks and run them parallelly across multiple CPUs, GPUs, TPUs, or other devices



Graph from *Hands-On Machine Learning with Scikit-Learn and TensorFlow*

Why graphs

1. Save computation. Only run subgraphs that lead to the values you want to fetch.
2. Break computation into small, differential pieces to facilitate auto-differentiation
3. Facilitate distributed computation, spread the work across multiple CPUs, GPUs, TPUs, or other devices
4. Many common machine learning models are taught and visualized as directed graphs

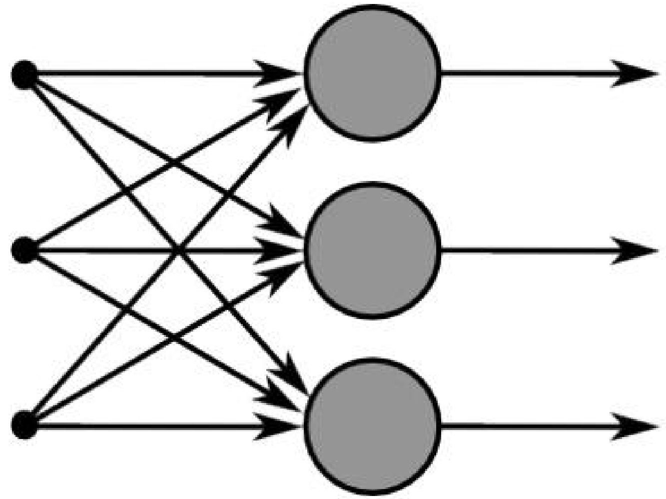


Figure 3: This image captures how multiple sigmoid units are stacked on the right, all of which receive the same input x.

Onward to … other knowledge representations

Jonathan Hudson, Ph.D.
jwhudson@ucalgary.ca
https://cspages.ucalgary.ca/~jwhudson/

UNIVERSITY OF CALGARY