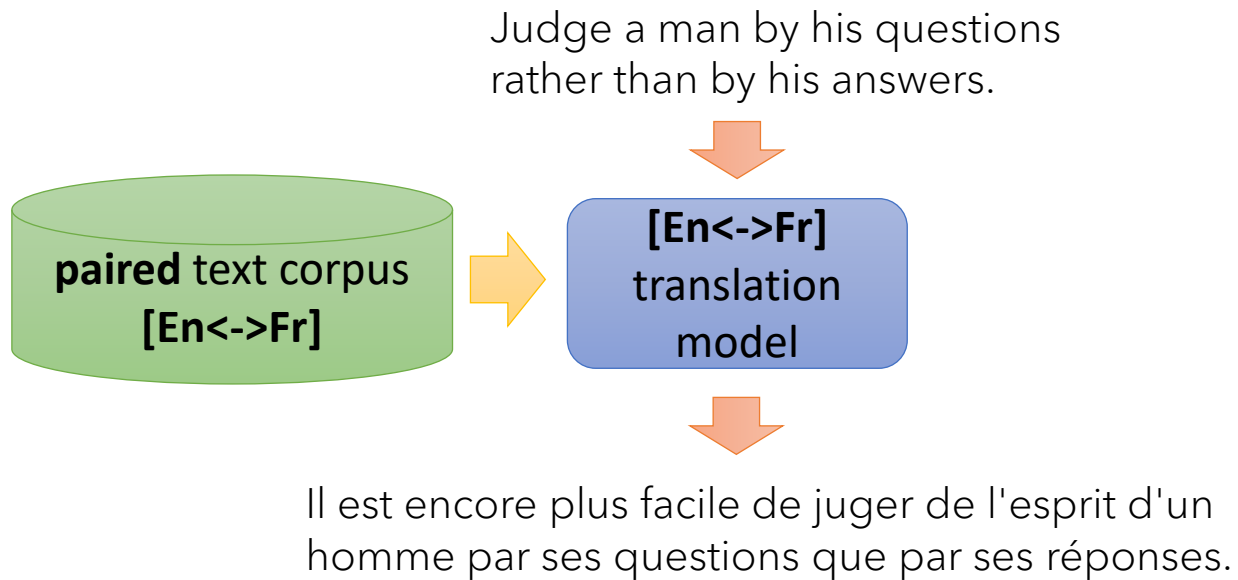


The world has over **6000** languages

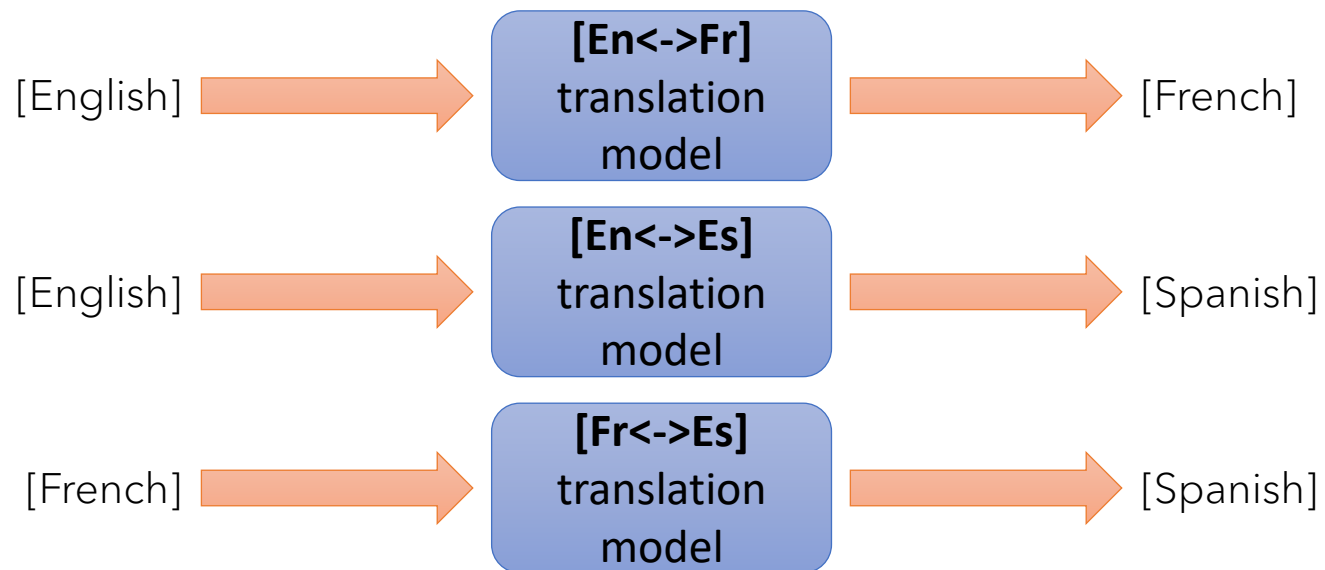
Automated translation systems require **paired data**

[En] I think, therefore I am. <-> [Fr] Je pense, donc je suis.

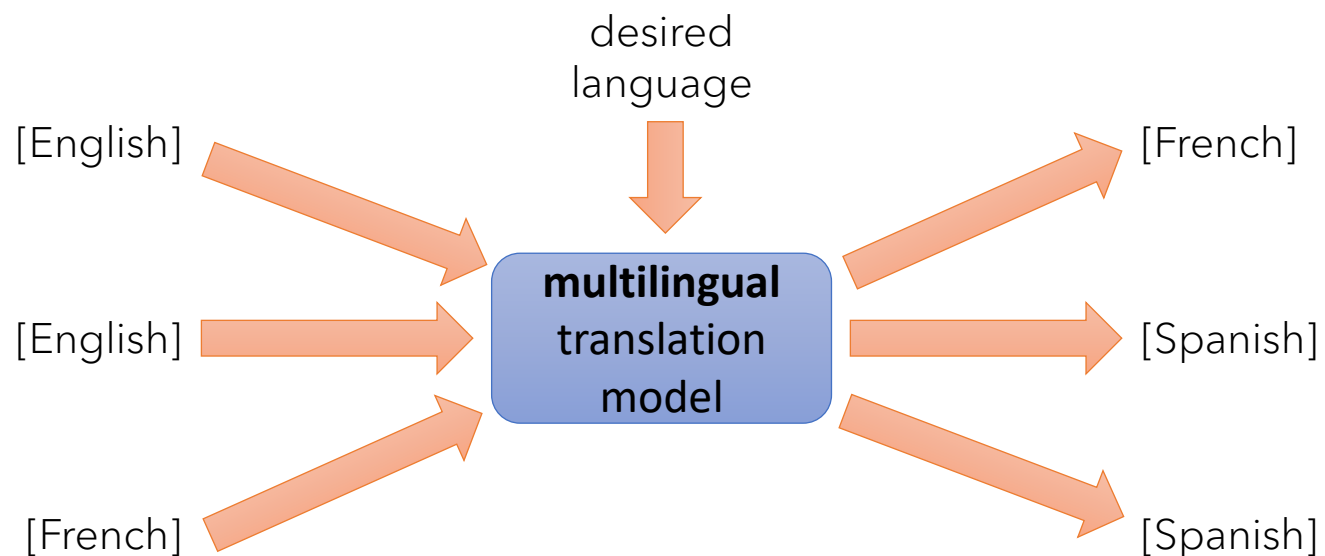


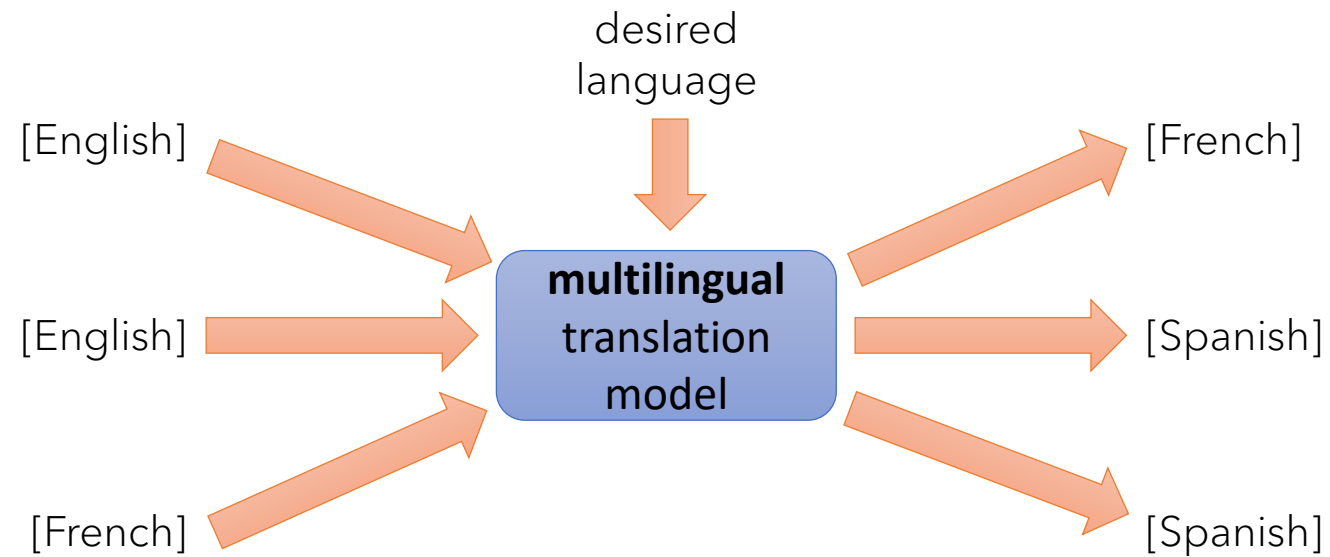
How many **paired** sentences are there for translating **Maltese** to **Tibetan**?

“Standard” machine translation:



“Multilingual” machine translation:





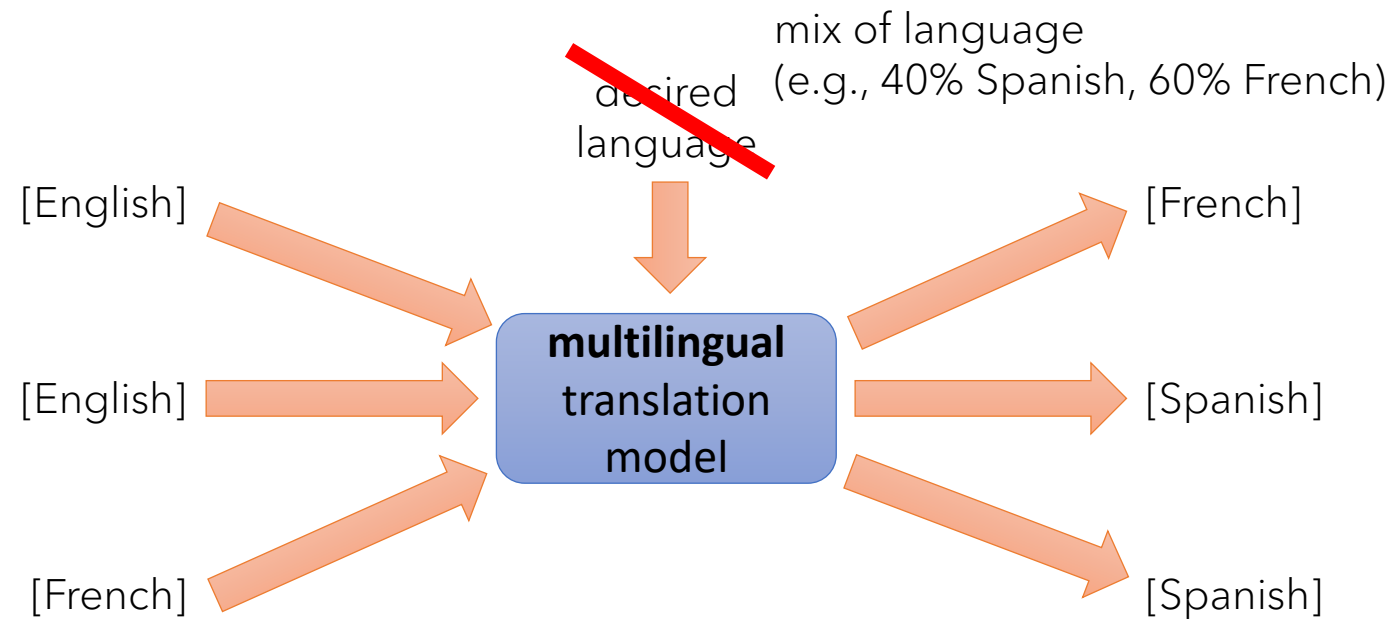
Improved efficiency:

Translating into and out of rare languages works **better** if the model is also trained on more common languages

What did they find?

Zero-shot machine translation:

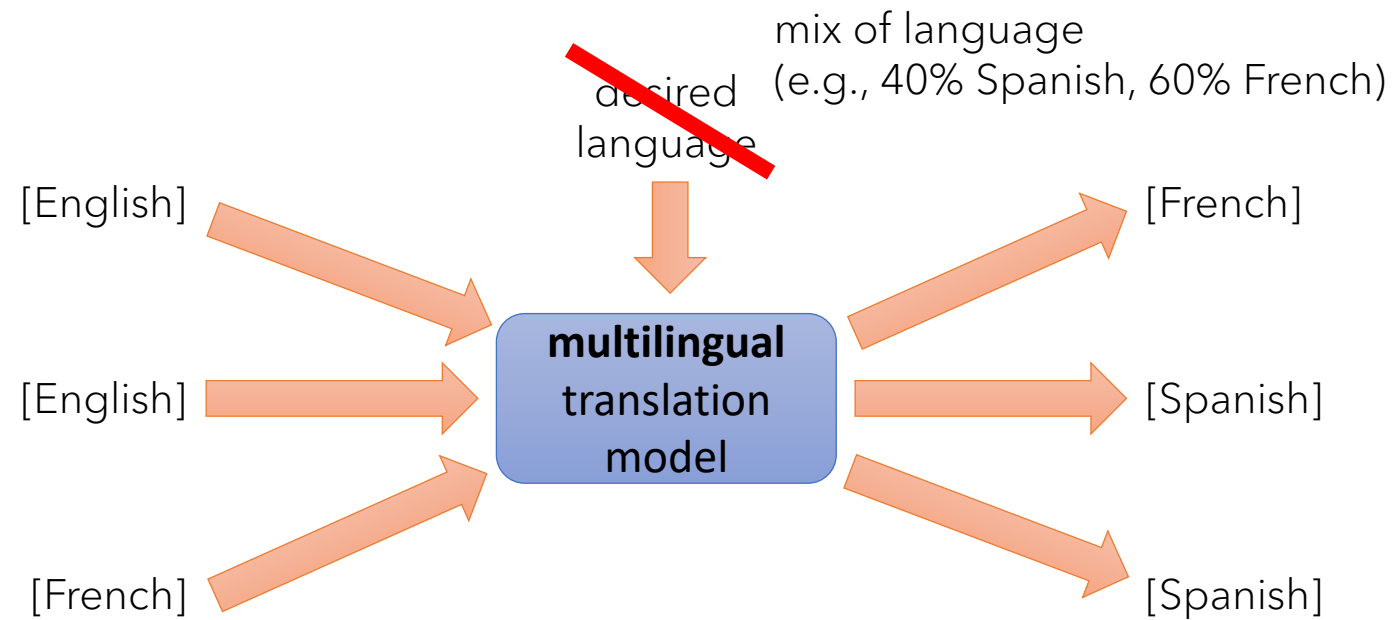
E.g., train on **English -> French**, **French -> English**, and **English -> Spanish**, and be able to translate **French -> Spanish**



Translating **English** to mix of **Spanish** and **Portuguese**:

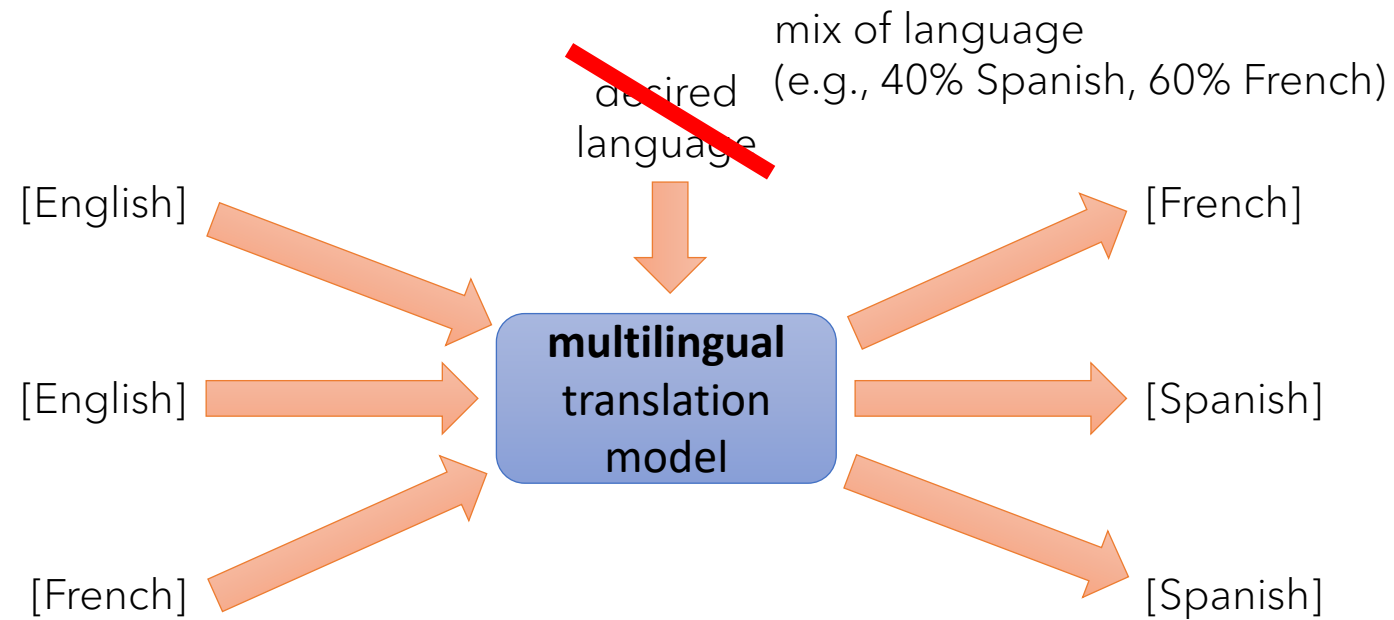
| Spanish/Portuguese: | Here the other guinea-pig cheered, and was suppressed. |
|---------------------|--|
| $w_{pt} = 0.00$ | Aquí el otro conejillo de indias animó, y fue suprimido. |
| $w_{pt} = 0.30$ | Aquí el otro conejillo de indias animó, y fue suprimido. |
| $w_{pt} = 0.40$ | Aquí, o outro porquinho-da-índia alegrou, e foi suprimido. |
| $w_{pt} = 0.42$ | Aqui o outro porquinho-da-índia alegrou, e foi suprimido. |
| $w_{pt} = 0.70$ | Aqui o outro porquinho-da-índia alegrou, e foi suprimido. |
| $w_{pt} = 0.80$ | Aqui a outra cobaia animou, e foi suprimida. |
| $w_{pt} = 1.00$ | Aqui a outra cobaia animou, e foi suprimida. |

“Portuguese” weight (Spanish weight = 1-w) →



Translating **English** to mix of **Japanese** and **Korean**:

| Japanese/Korean: | I must be getting somewhere near the centre of the earth. |
|------------------|---|
| $w_{ko} = 0.00$ | 私は地球の中心の近くはどこかに行っているに違いない。 |
| $w_{ko} = 0.40$ | 私は地球の中心近くのどこかに着いているに違いない。 |
| $w_{ko} = 0.56$ | 私は地球の中心の近くのところになっているに違いない。 |
| $w_{ko} = 0.58$ | 私は 지구 中心의 가까이 어딘가에도 착하고 있어야 한다. |
| $w_{ko} = 0.60$ | 나는 지구의 센터의 가까이 어딘가에도 착하고 있어야 한다. |
| $w_{ko} = 0.70$ | 나는 지구의 중심 근처 어딘가에도 착해야 합니다. |
| $w_{ko} = 0.90$ | 나는 어딘가 지구의 중심 근처에도 착해야 합니다. |
| $w_{ko} = 1.00$ | 나는 어딘가 지구의 중심 근처에도 착해야 합니다. |

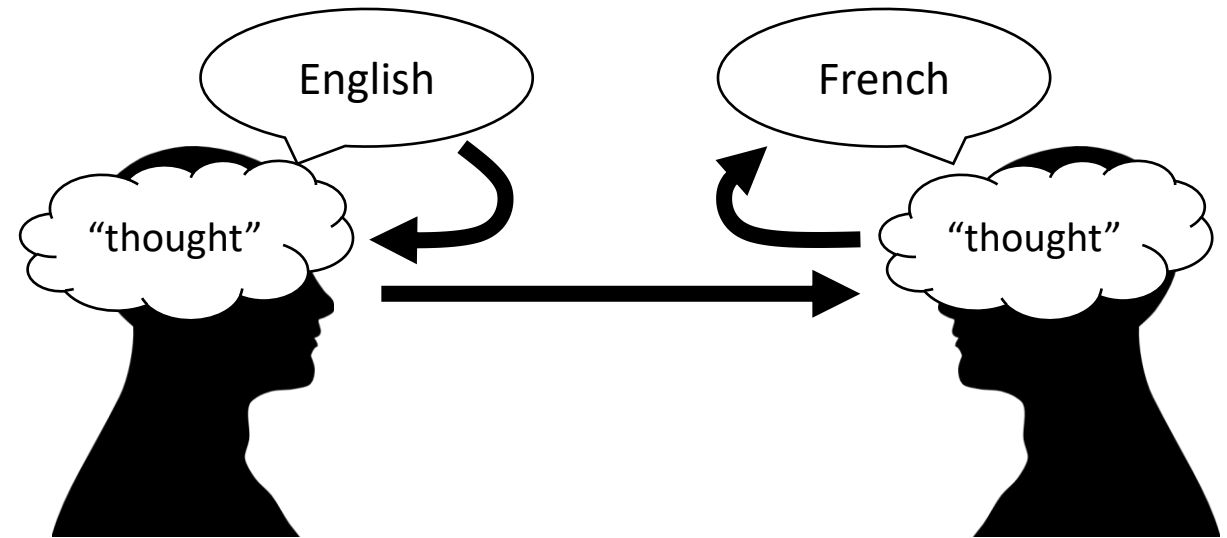
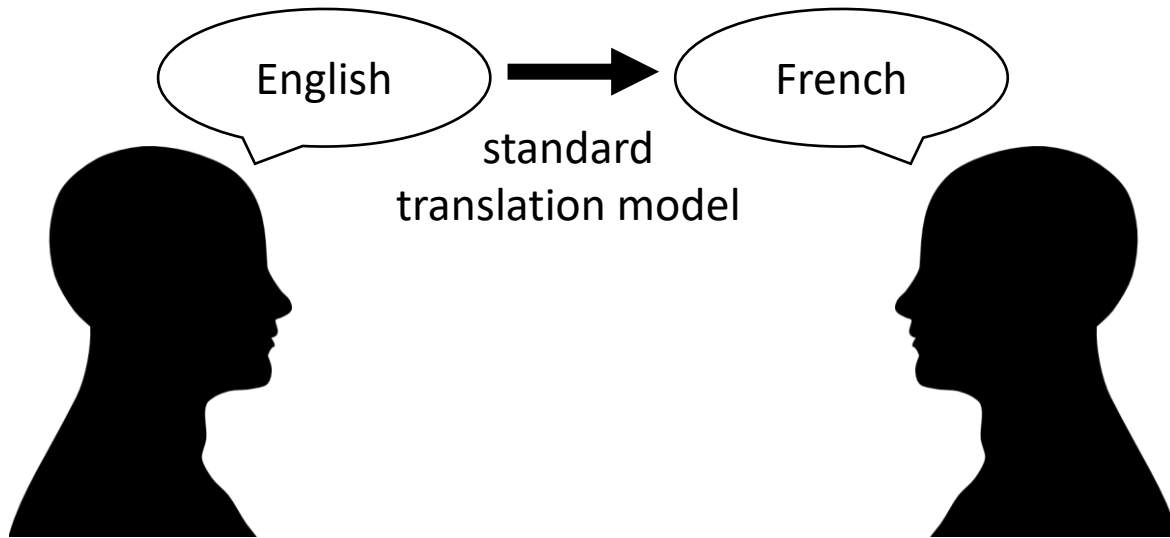


Translating **English** to mix of **Russian** and **Belarusian**:

| Russian/Belarusian: | I wonder what they'll do next! |
|---------------------|--|
| $w_{be} = 0.00$ | Интересно, что они сделают дальше! |
| $w_{be} = 0.20$ | Интересно, что они сделают дальше! |
| $w_{be} = 0.30$ | <u>Цікаво</u> , что они будут делать дальше! |
| $w_{be} = 0.44$ | <u>Цікаво</u> , що вони будуть робити далі! |
| $w_{be} = 0.46$ | <u>Цікаво</u> , що вони будуть робити далі! |
| $w_{be} = 0.48$ | <u>Цікаво</u> , што яны зробяць далей! |
| $w_{be} = 0.50$ | Цікава, што яны будуць рабіць далей! |
| $w_{be} = 1.00$ | Цікава, што яны будуць рабіць далей! |

Neither Russian nor Belarusian!

What's going on?

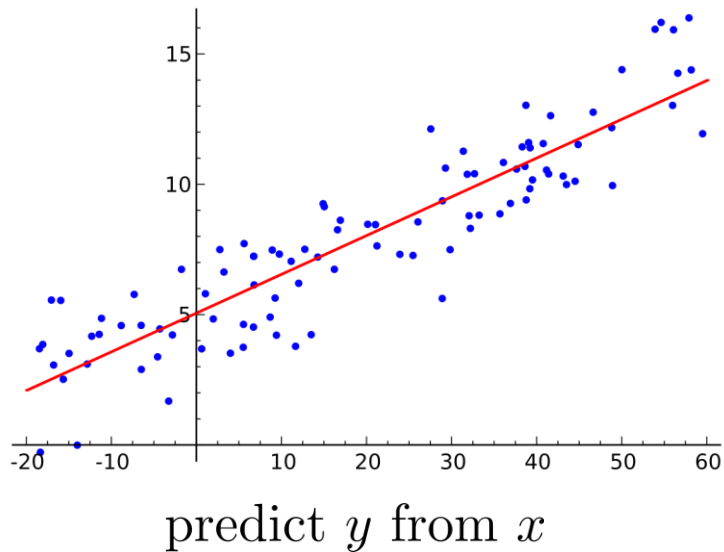


the "thought" is a **representation!**

Representation learning

Handling such complex inputs requires **representations**

“Classic” view of machine learning:

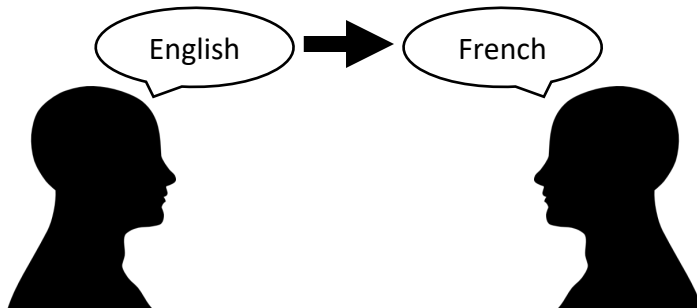
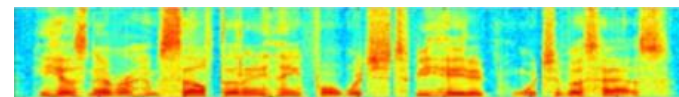


but what is x ?

Il est encore plus facile de juger de l'esprit d'un homme par ses questions que par ses réponses.



The power of deep learning lies in its ability to **learn** such **representations** automatically from data



Deep Learning

Designing, Visualizing and Understanding Deep Neural Networks

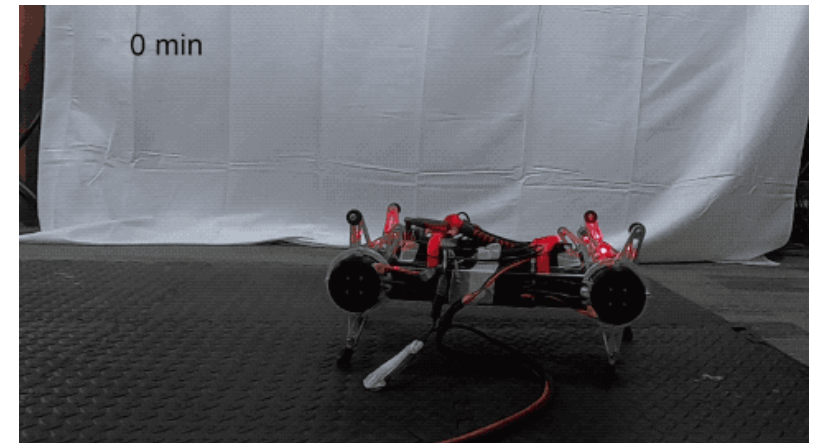
CS W182/282A

Instructor: Sergey Levine
UC Berkeley



Course overview

- **Broad** overview of deep learning topics
 - Neural network architectures
 - Optimization algorithms
 - Applications: vision, NLP
 - Reinforcement learning
 - Advanced topics
- **Four** homework programming assignments
 - Neural network basics
 - Convolutional and recurrent networks
 - Natural language processing
 - Reinforcement learning
- **Two** midterm exams
 - Format TBD, but most likely will be a take-home exam
- **Final project** (group project, 2-3 people)
 - Most important part of the course
 - CS182: choose vision, NLP, or reinforcement learning
 - CS282: self-directed and open-ended project



Course policies

Grading:

30% midterms
40% programming homeworks
30% final project

Late policy:

5 slip days
strict late policy, no slack beyond slip days
no slip days for final project (due to grades deadline)

Prerequisites:

Excellent knowledge of calculus linear algebra

especially: multi-variate derivatives, matrix operations, solving linear systems

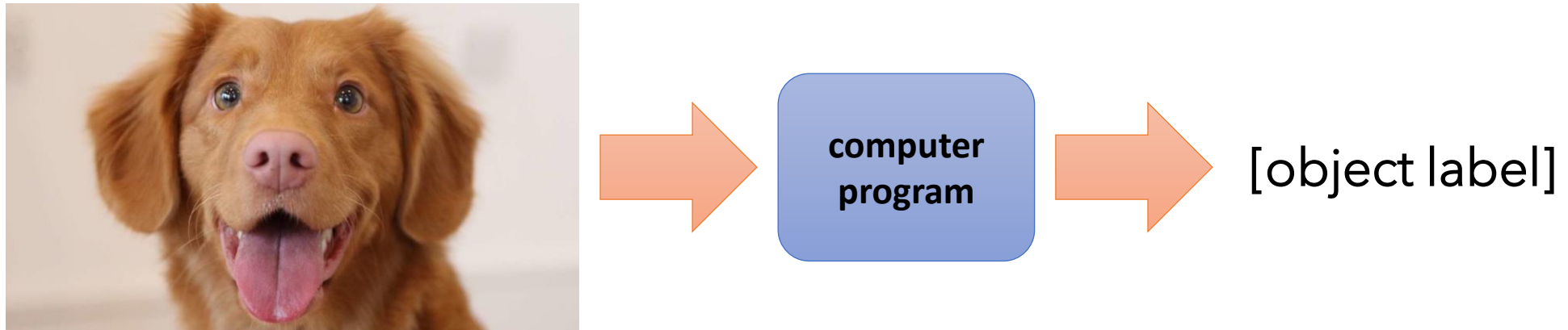
CS70 or STAT134, excellent knowledge of probability theory (including continuous random variables)

CS189, or a very strong statistics background

CS61B or equivalent, able to program in Python

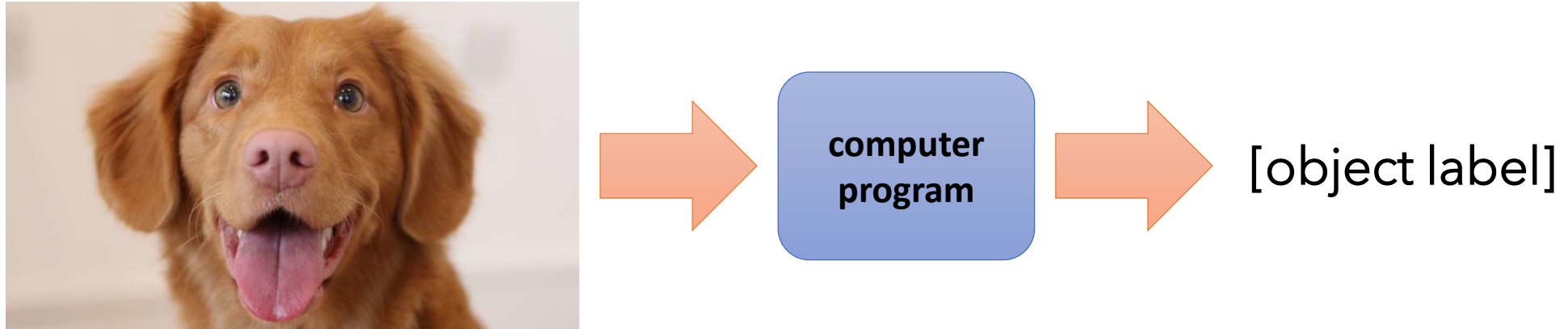
What is machine learning?
What is deep learning?

What is machine learning?



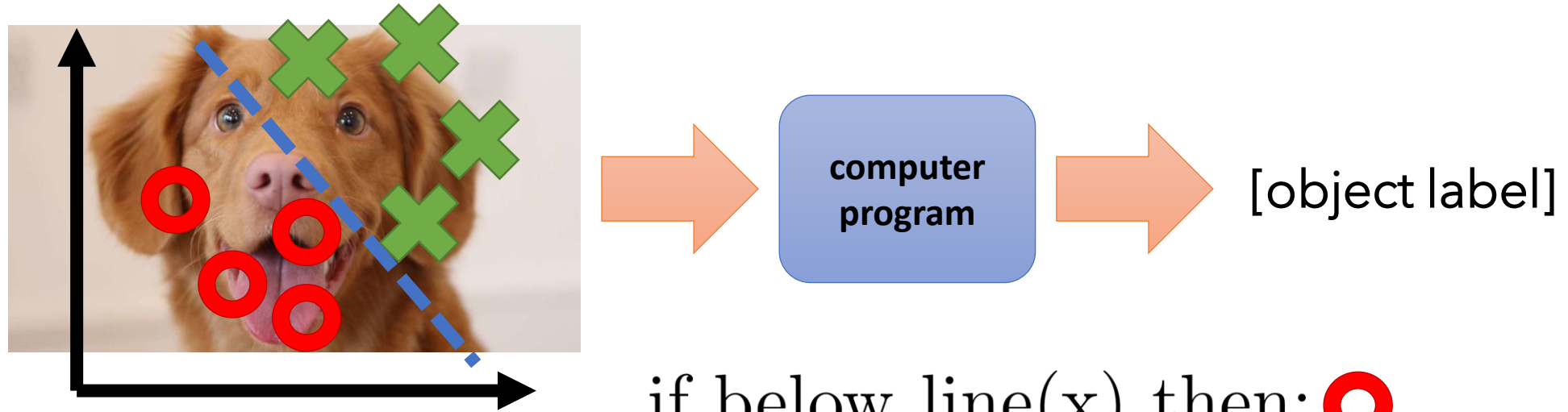
- How do we implement this program?
- A function is a set of **rules** for transforming **inputs** into **outputs**
- Sometimes we can define the rules by hand – this is called programming
- What if we don't know the rules?
- What if the rules are too complex? Too many exceptions & special cases?


What is machine learning?



- Instead of defining the **input** -> **output** relationship by hand, define a program that acquires this relationship from **data**
- **Key idea:** if the rules that describe how **inputs** map to **outputs** are complex and full of special cases & exceptions, it is easier to provide **data** or **examples** than to implement those rules
- **Question:** Does this also apply to human and animal learning?

What are we learning?



if below_line(x) then: 

else:  

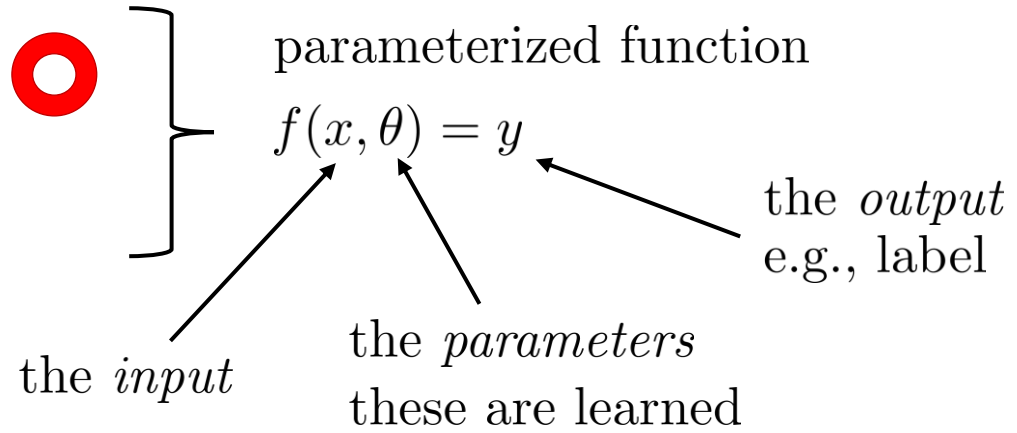
this describes a line $\longrightarrow x_1\theta_1 + x_2\theta_2 + \theta_3 \leq 0$

learn $(\theta_1, \theta_2, \theta_3) = \vec{\theta}$ $\vec{x}^T \vec{\theta} \leq 0$

so that our *parameterized* program (function) gives the right answer!

In general...

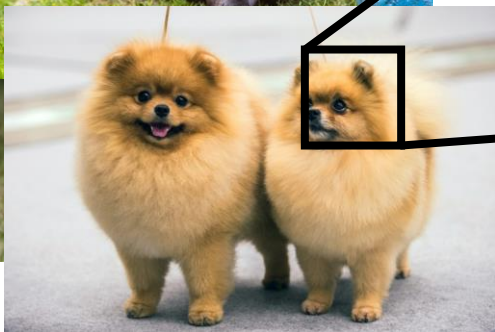
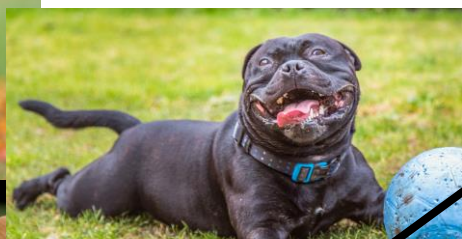
if below_line(x) then: ○
else: ✕



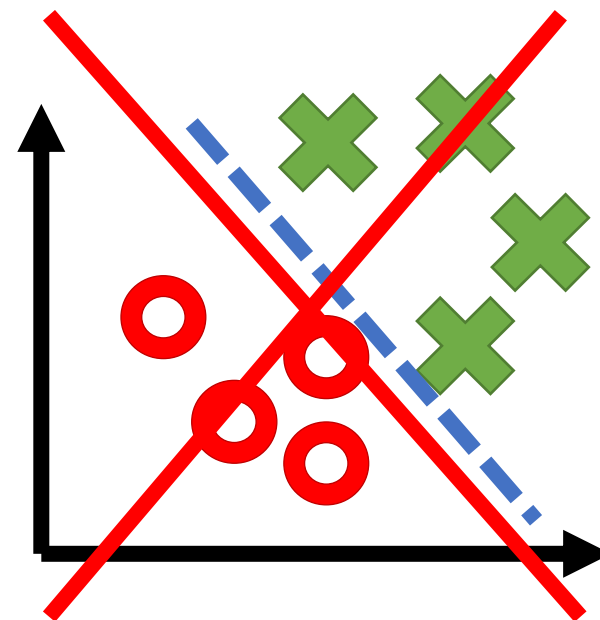
can also write as: $f_{\theta}(x) = y$

crucially, $f_{\theta}(x)$ can be almost any expression of x and θ !

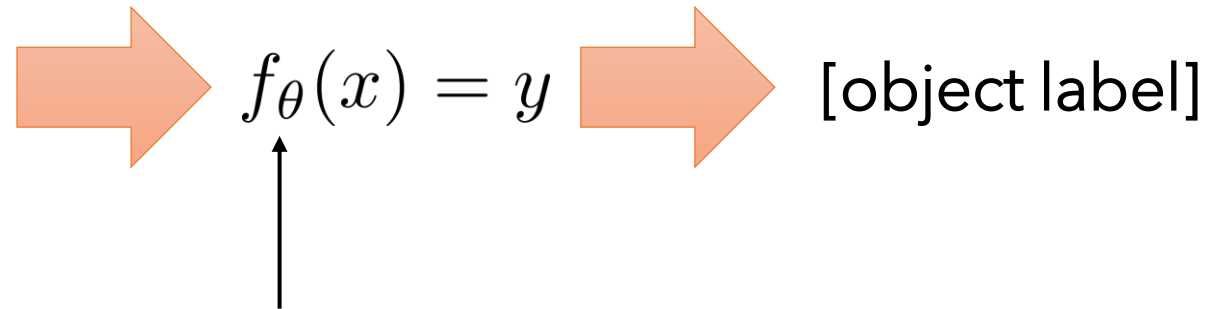
But what parameterization do we use?



| | | | |
|-----|-----|-----|-----|
| 0.2 | 0.1 | 0.3 | 0.3 |
| 0.2 | 0.5 | 0.3 | 0.3 |
| 0.3 | 0.1 | 0.2 | 0.2 |
| 0.3 | 0.1 | 0.2 | 0.2 |

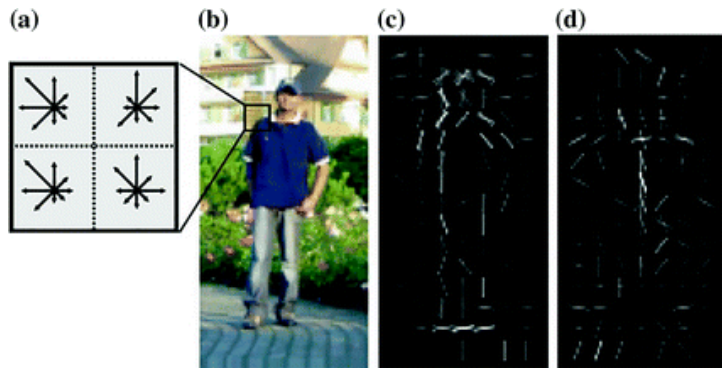


“Shallow” learning



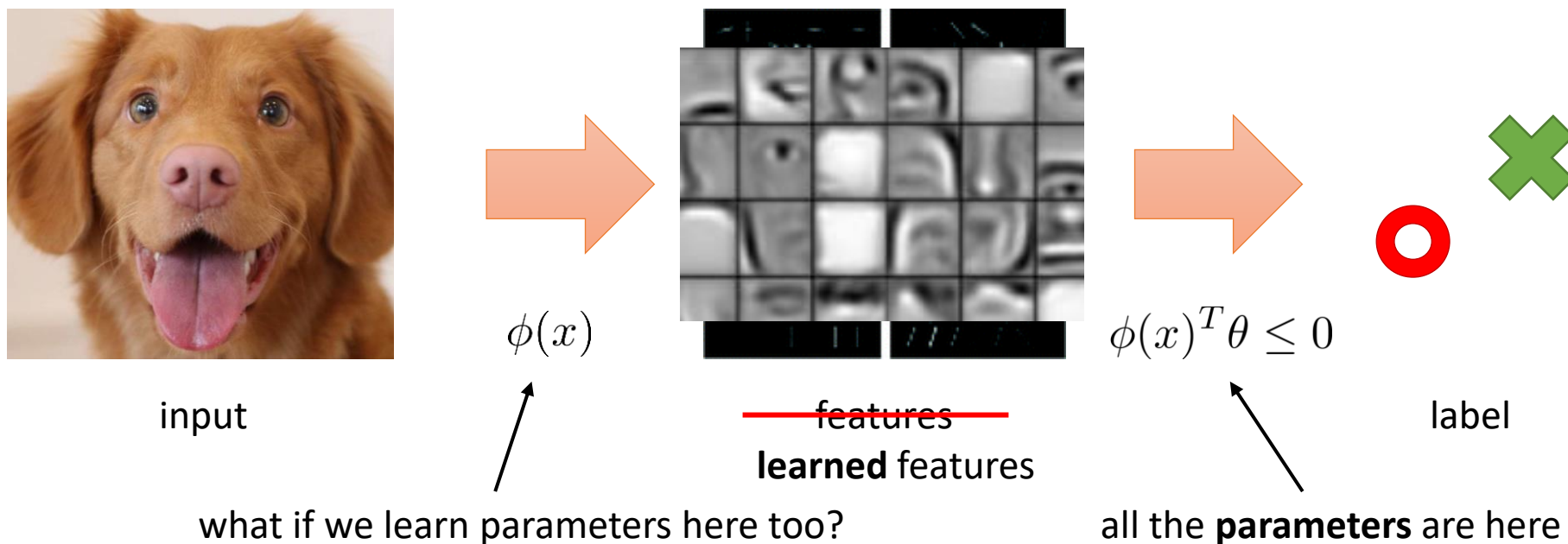
fixed function for extracting *features* from x

$$\phi(x)^T \theta \leq 0$$

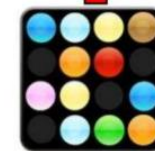
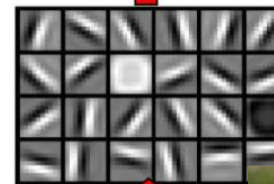
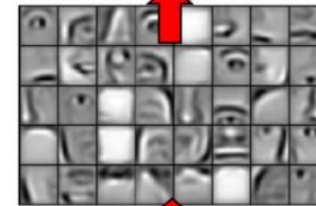
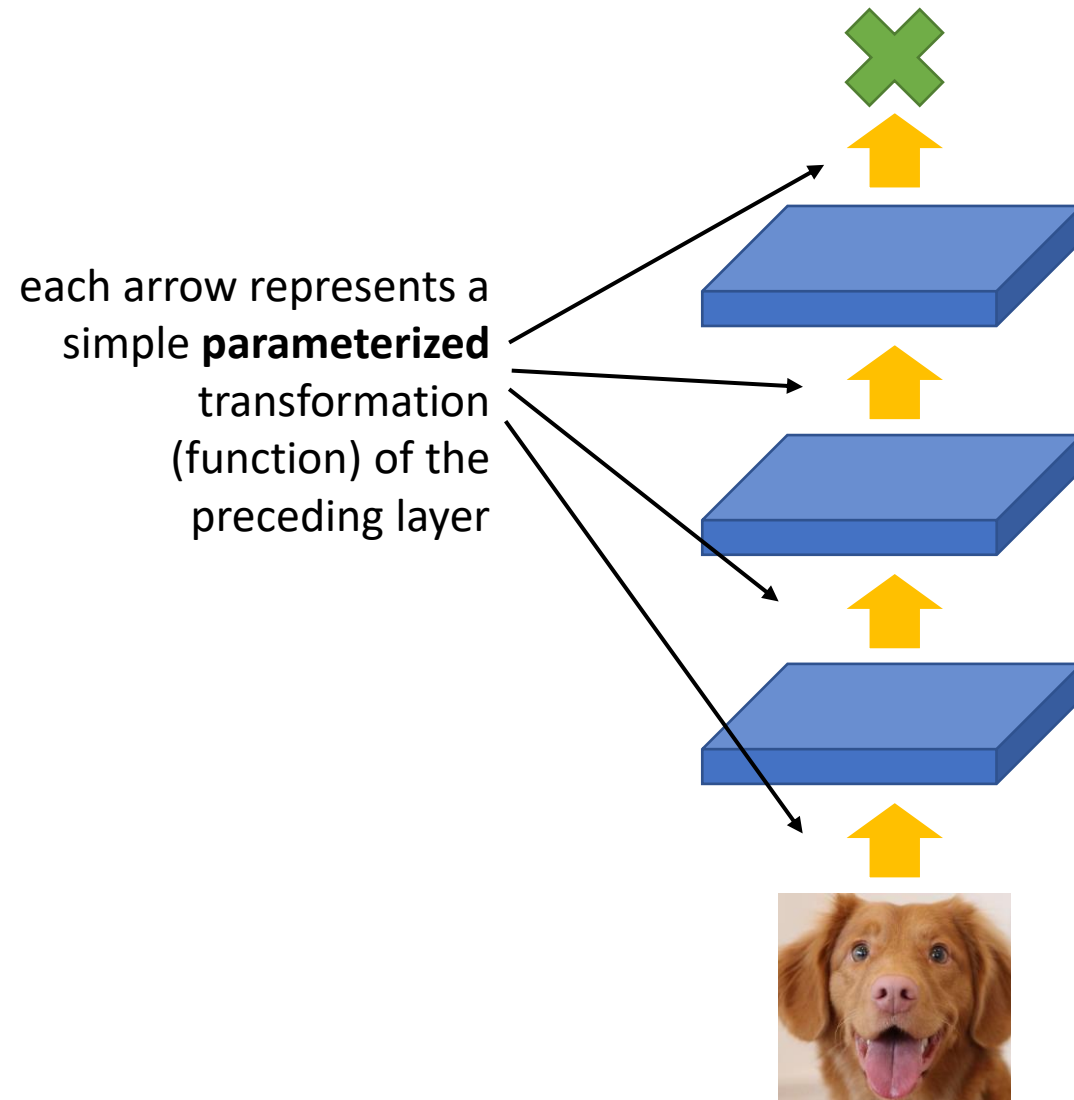


- Kind of a “compromise” solution: don’t hand-program the rules, but hand-program the features
- Learning on top of the features can be simple (just like the 2D example from before!)
- Coming up with good features is very hard!

From shallow learning to deep learning

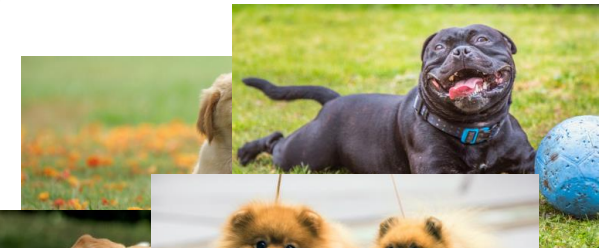


Multiple layers of representations?

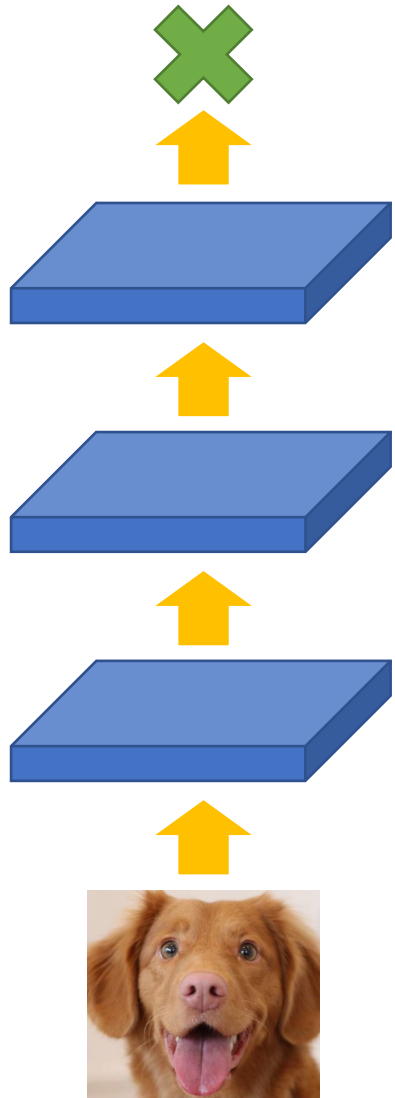


Higher level representations are:

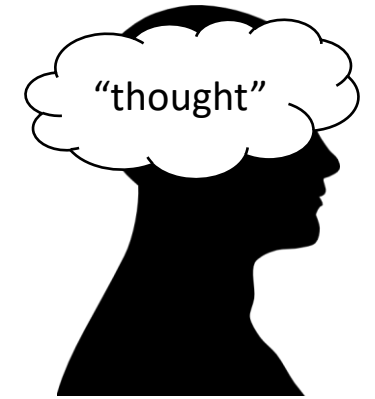
- More abstract
- More invariant to nuisances
- Easier for predicting label



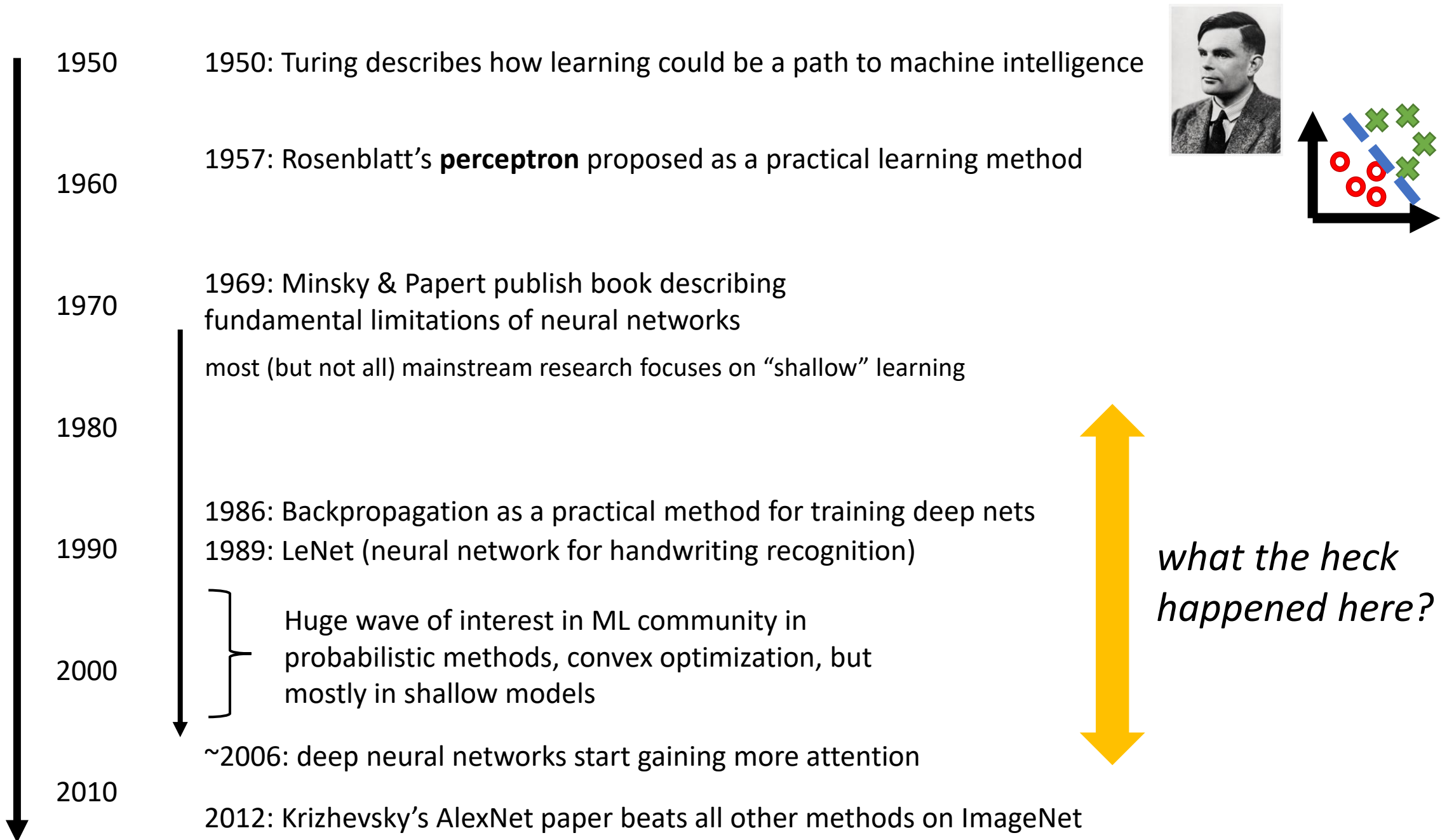
So, what is deep learning?



- Machine learning with **multiple layers** of **learned representations**
- The **function** that represents the transformation from input to internal representation to output is usually a deep neural network
 - This is a bit circular, because almost all **multi-layer parametric** functions with **learned parameters** can be called neural networks (more on this later)
- The parameters for every layer are usually (**but not always!**) trained with respect to the overall task objective (**e.g., accuracy**)
 - This is sometimes referred to as **end-to-end** learning

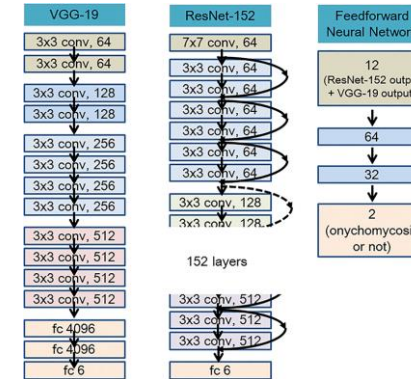


What makes deep learning work?



What makes deep learning work?

1) **Big** models with many layers



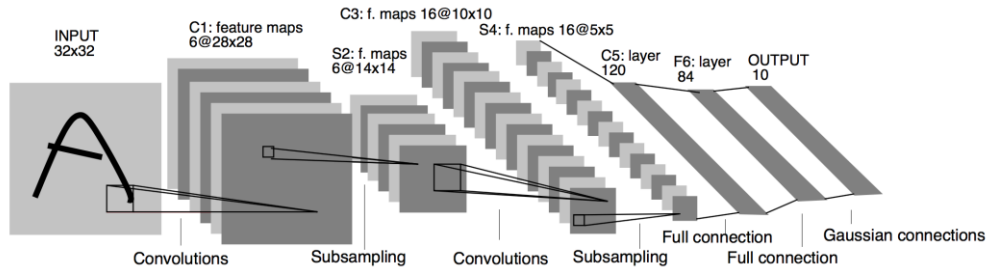
2) **Large** datasets with many examples



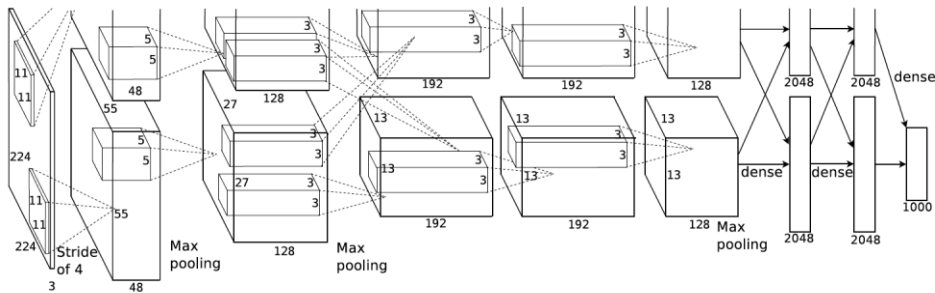
3) Enough **compute** to handle all this



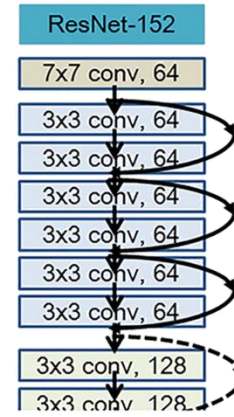
Model scale: is more layers better?



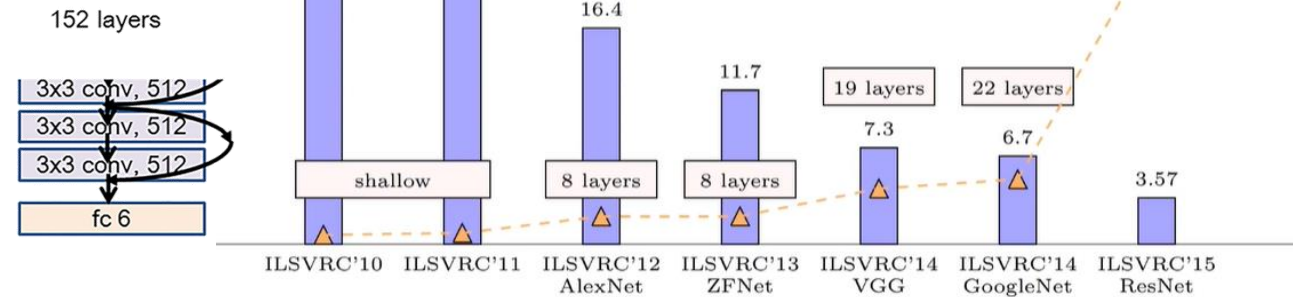
LeNet, 7 layers (1989)



Krizhevsky's model (AlexNet) for ImageNet, 8 layers (2012)

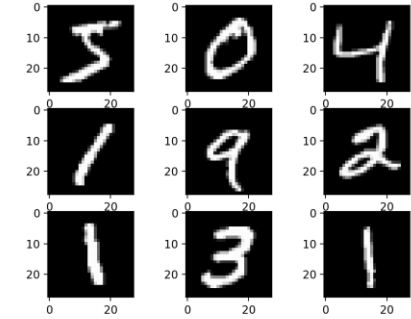


ResNet-152: 152 layers (2015)



How big are the datasets?

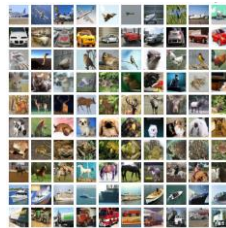
MNIST (handwritten characters), 1990s - today: 60,000 images



CalTech 101, 2003: ~9,000 images



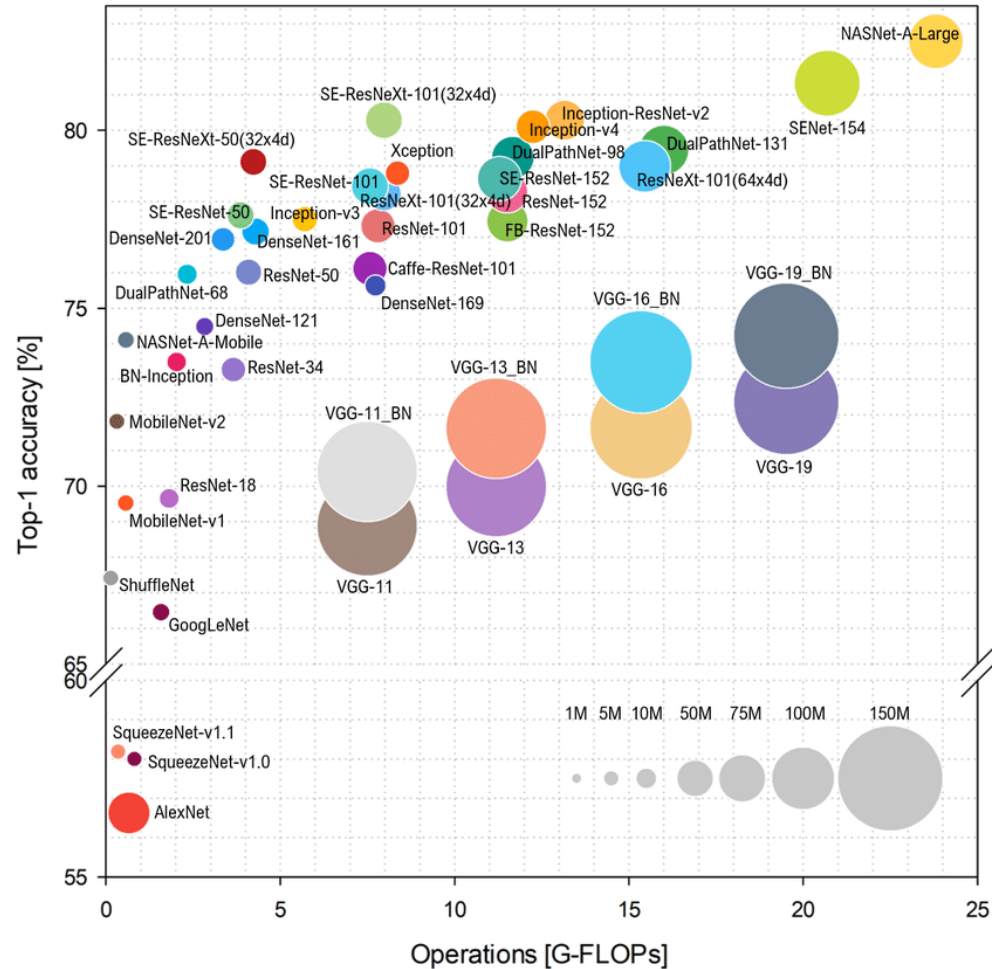
CIFAR 10, 2009: ~60,000 images



ILSVRC (ImageNet), 2009: 1.5 million images



How does it scale with compute?



What about NLP?

how long does it take to train BERT

All News Shopping Images

About 21,700,000 results (0.78 seconds)

about 54 hours

On what?? on this:

about 16 TPUs
(this photo shows a few
thousand of these)



So... it's really expensive?

- **One perspective:** deep learning is not such a good idea, because it requires huge models, huge amounts of data, and huge amounts of compute
- **Another perspective:** deep learning is great, because as we add more data, more layers, and more compute, the models get better and better!



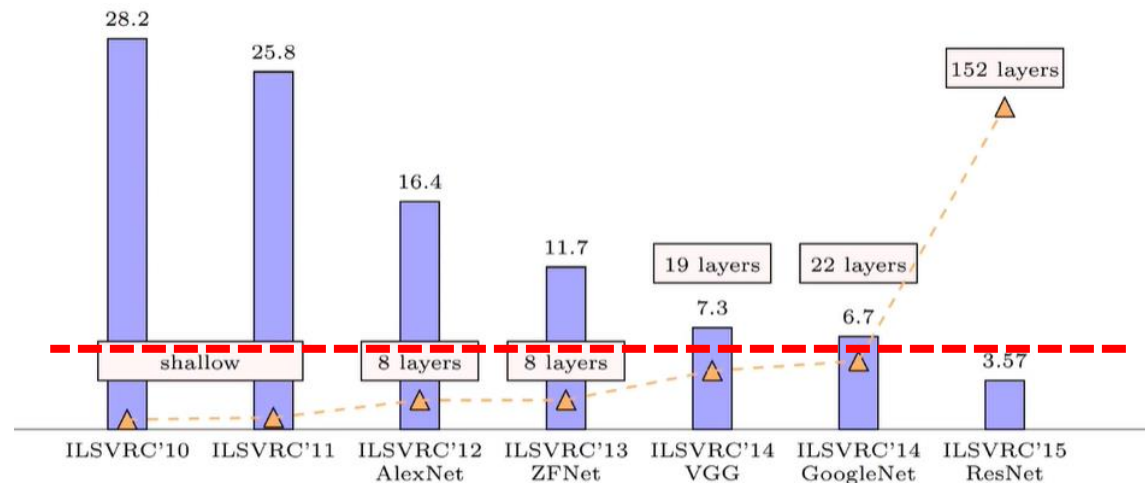
...which human?

 Andrej Karpathy blog

[About](#)

What I learned from competing against a ConvNet on ImageNet

Sep 2, 2014



human performance:
about 5% error

The underlying themes

- **Acquire representations** by using **high-capacity** models and lots of **data**, without requiring manual engineering of features or representations
 - Automation: we don't need to **know** what the good features are, we can have the model figure it out from data
 - Better performance: when representations are learned end-to-end, they are better tailored to the current task
- **Learning vs. inductive bias** ("nature vs. nurture"): models that get most of their performance from their data rather than from designer insight
 - **Inductive bias**: what we build into the model to make it learn effectively (we can never fully get rid of this!)
 - Should we build in **knowledge**, or better machinery for learning and scale?
- **Algorithms that scale**: This often refers to methods that can get better and better as we add more data, representational capacity, and compute

Model capacity: (informally) how many different functions a particular model class can represent (e.g., all linear decision boundaries vs. non-linear boundaries).

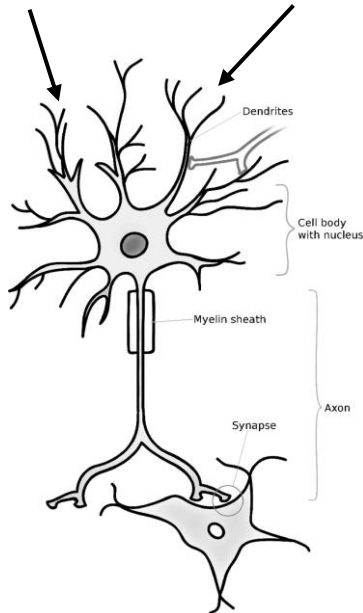
Inductive bias: (informally) built-in knowledge or biases in a model designed to help it learn. All such knowledge is "bias" in the sense that it makes some solutions more likely and some less likely.

Scaling: (informally) ability for an algorithm to work better as more data and model capacity is added.

Why do we call them **neural** nets?

Early on, neural networks were proposed as a rudimentary model of neurons in the brain

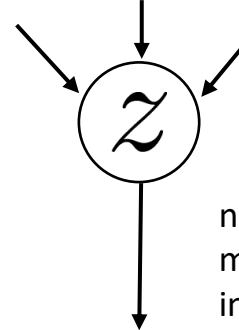
dendrites receive signals from other neurons



neuron “decides” whether to fire based on incoming signals

axon transmits signal to downstream neurons

artificial “neuron” sums up signals from upstream neurons (also referred to as “units”)



neuron “decides” how much to fire based on incoming signals

activations transmitted to downstream units

$$z = \sum_i a_i$$

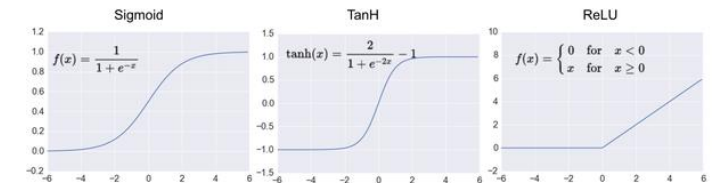
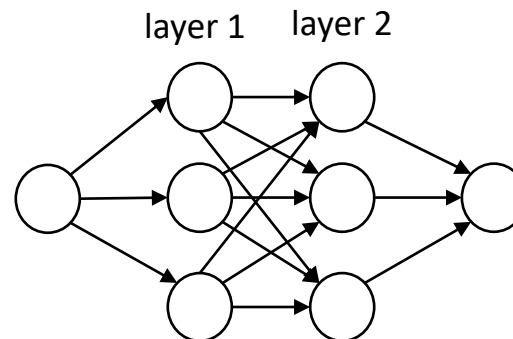
upstream activations

$$a = \sigma(z)$$

“activation function”

Is this a good model for real neurons?

- Crudely models *some* neuron function
- Missing many other important anatomical details
- Don’t take it too seriously



What does deep learning have to do with the brain?

Unsupervised learning models of primary cortical receptive fields and receptive field plasticity

Andrew Saxe, Maneesh Bhand, Ritvik Mudur, Bipin Suresh, Andrew Y. Ng
Department of Computer Science
Stanford University
{asaxe, mbhand, rmudur, bipins, ang}@cs.stanford.edu

Does this mean that the brain does deep learning?

Or does it mean that any sufficiently powerful learning machine will basically derive the same solution?

