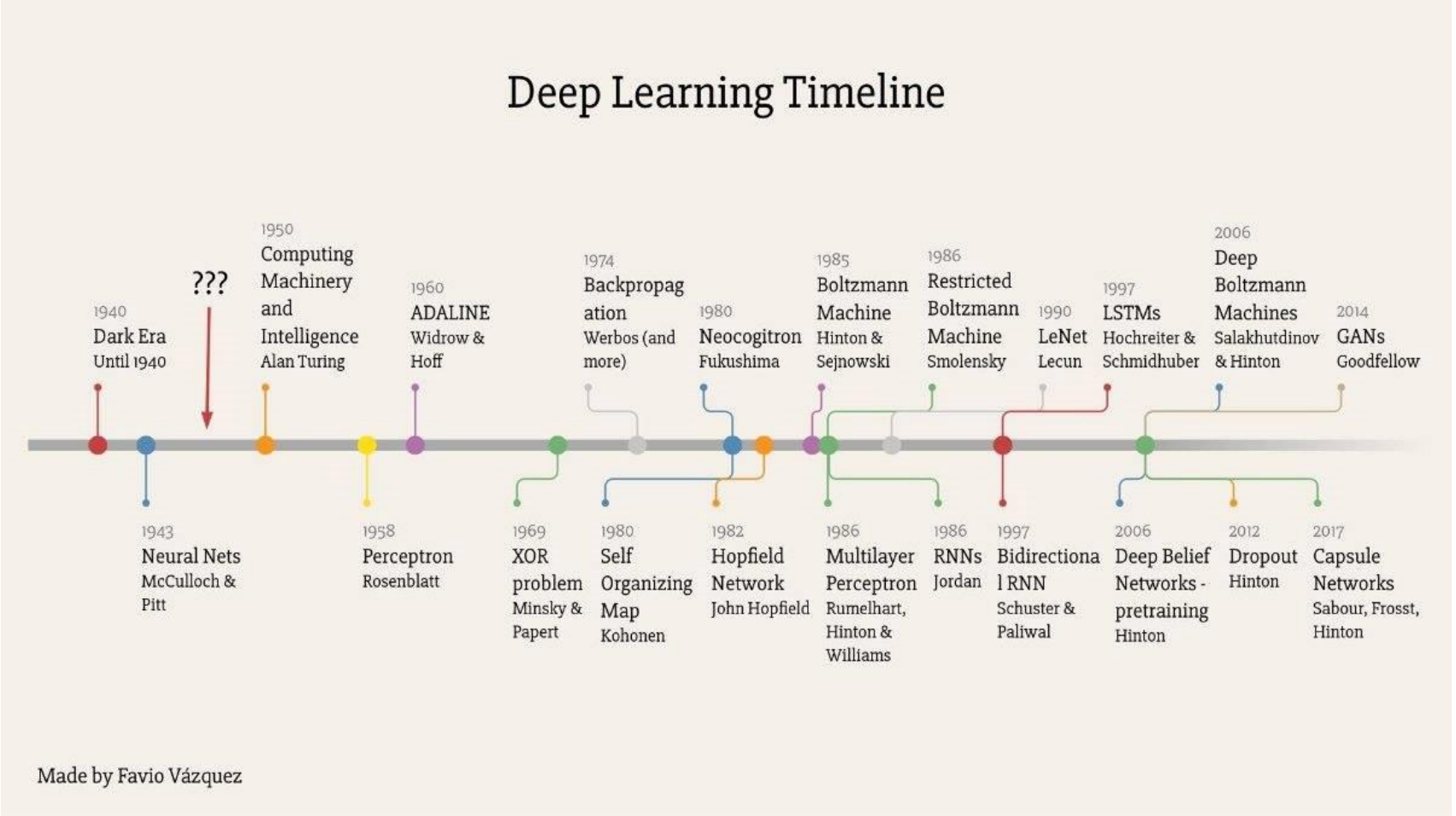
02476 Machine Learning Operations

Nicki Skafte Detlefsen

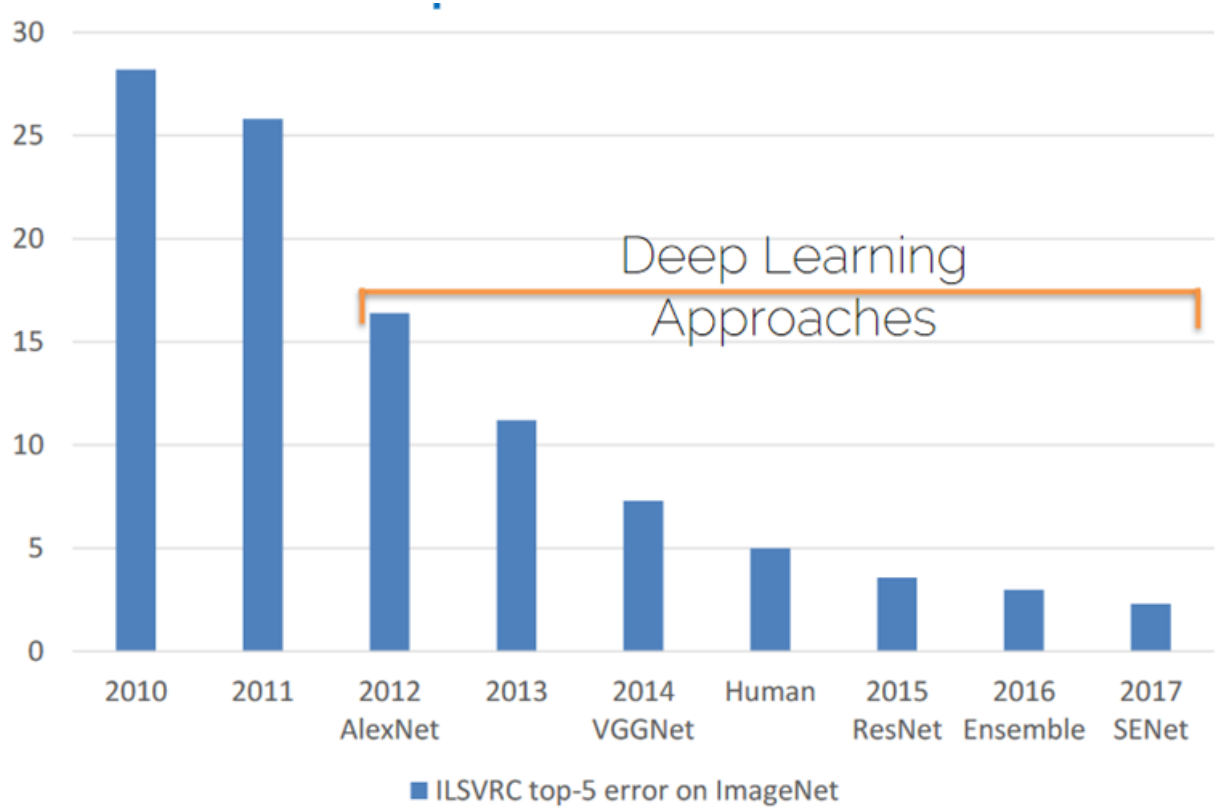
Deep learning software

Deep learning is old



The deep learning revolution

- In 2012 the Alexnet paper crushed all competition on in the imagenet benchmark



Since then

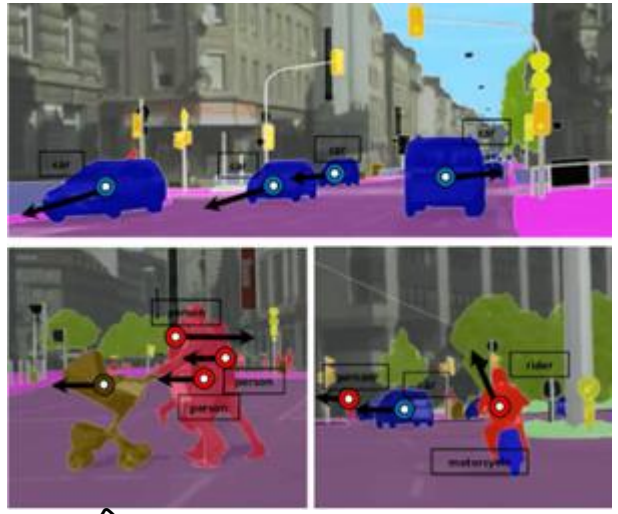
```
Q: What is your favorite animal?
A: My favorite animal is a dog.

Q: Why?
A: Because dogs are loyal and friendly.

Q: What are two reasons that a dog might be in a bad mood?
A: Two reasons that a dog might be in a bad mood are if it is hungry or if it is hot.

Q: How many eyes does a giraffe have?
A: A giraffe has two eyes.

Q: How many legs does a frog have?
A: A frog has four legs.

Q: Are there any animals with three legs?
A: No, there are no animals with three legs.

Q: Why don't animals have three legs?
A: Animals don't have three legs because they would fall over.
```

- ChatGPT:

Generation of text

- AlphaGO:

Beating humans at their own games

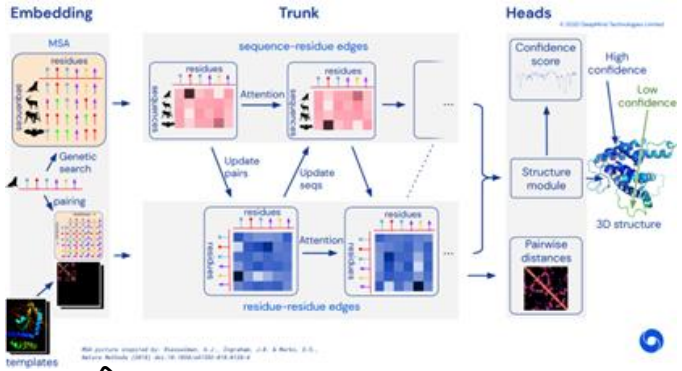


- U-net:

Real-time semantic segmentation

- Stable diffusion:

Text to image translation/generation

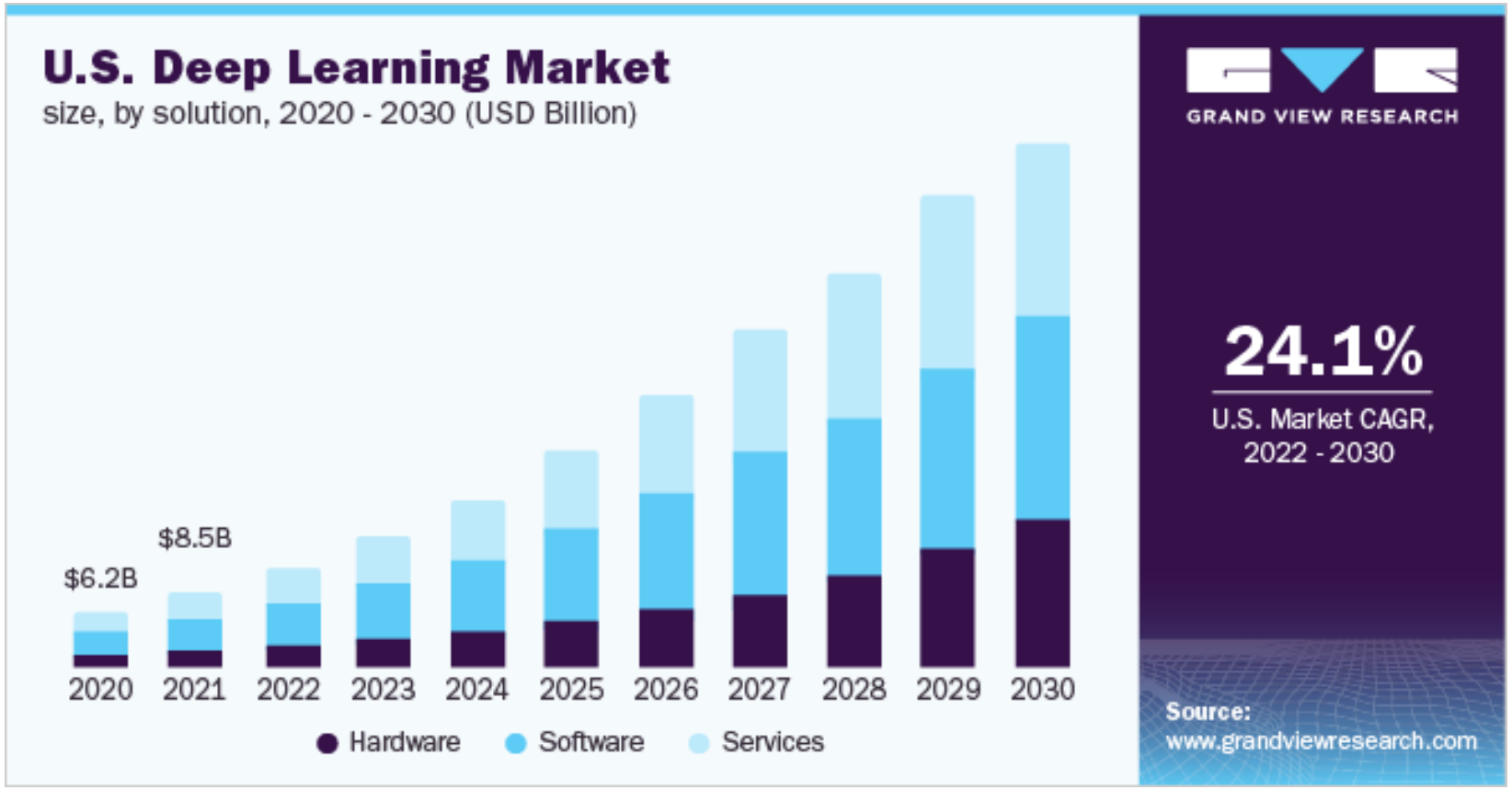


- Alphafold:

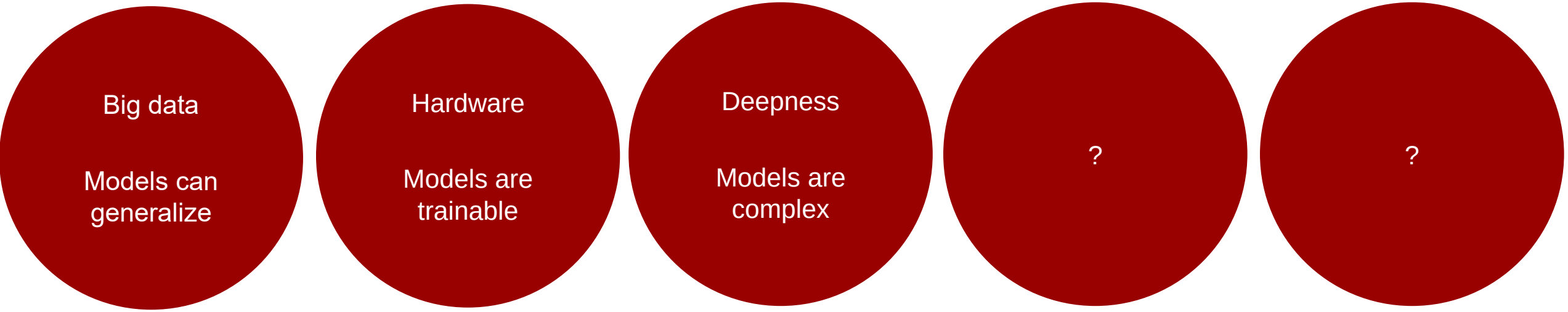
Solving protein engineering

Why you should care about deep learning

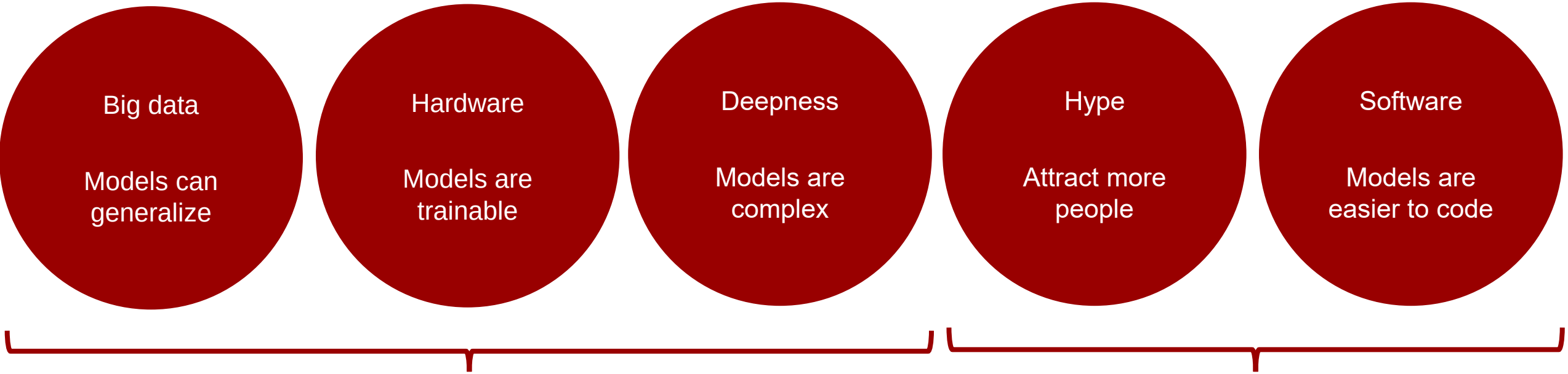
- Usage of Deep Learning in industry is increasing very fast!



The drivers of the revolution



The drivers of the revolution



Why do need specialized frameworks for DL?

Deep learning is just a log of simple math

💡 But we need to do it efficiently

💡 We need to take care of hardware acceleration (e.g. it can run on GPU)

💡 We need to take care of gradient backpropgation

💡 Optimizers, data interface etc. also complicates implementation

We do not want to deal with this ourself ⚠

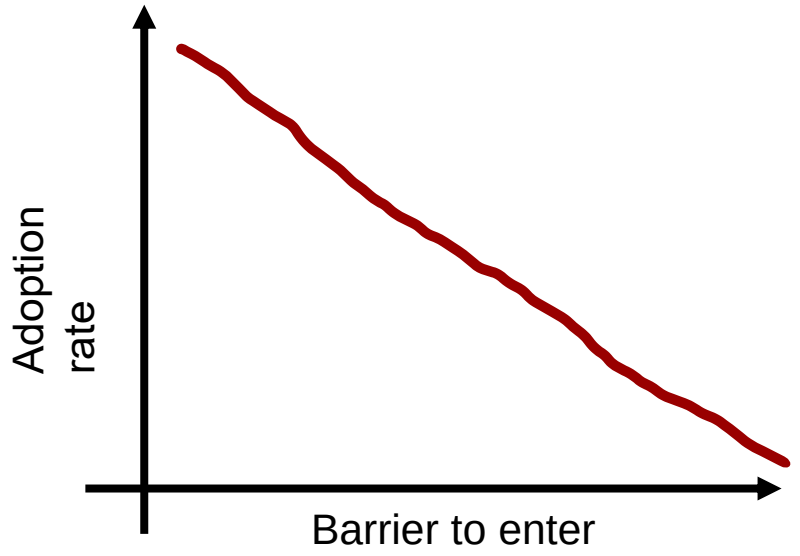
```python
import numpy as np

class Linear(object):
    def __init__(self, input_dim: int, num_hidden: int = 1):
        self.weight = np.random.randn(input_dim, num_hidden)
        self.bias = np.zeros(num_hidden)

    def __call__(self, x):
        self.x = x
        output = x @ self.weight + self.bias
        return output

    def backward(self, gradient):
        self.weight_gradient = self.x.T @ gradient
        self.bias_gradient = gradient.sum(axis=0)
        self.x_gradient = gradient @ self.weight
        return self.x_gradient

    def update(self, lr):
        self.weight = self.weight - lr * self.weight_gradient
        self.bias = self.bias - lr * self.bias_gradient


if __name__ == "__main__":
    x = np.random.randn(10, 5)
    layer = Linear(5, 1)
    y = layer(x)
    grad = layer.backward(np.ones((10, 5)))
    layer.update(1e-2)
```

Barrier to enter

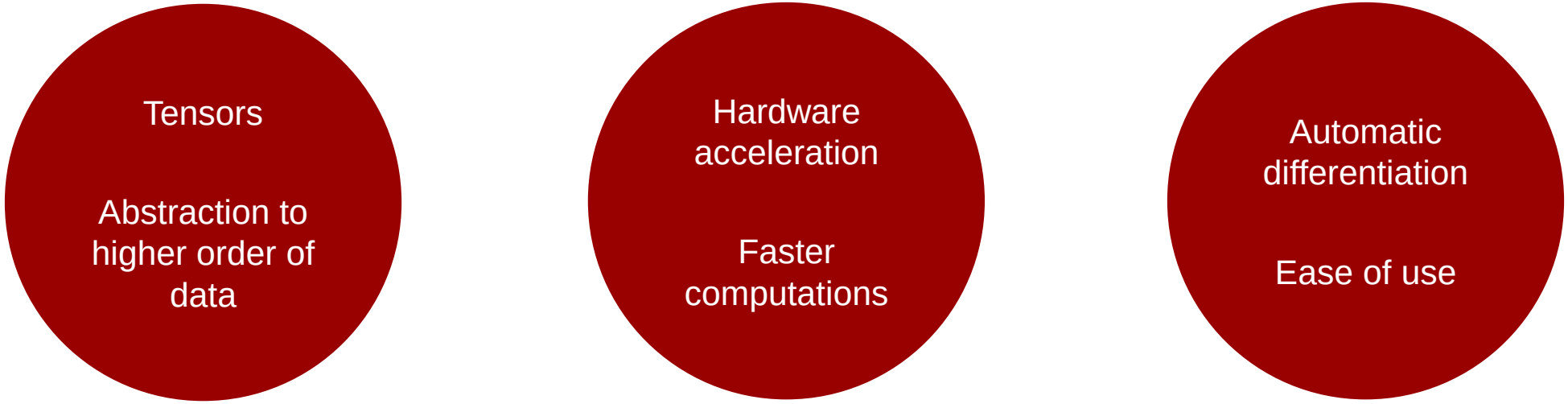
- Without proper DL frameworks, ML/DL/AI as a field would have a very high barrier to enter

- Low barrier to enter mean it accessible to more people = more people driving the technology forward

- AI would be gated from the public (trustworthy AI)



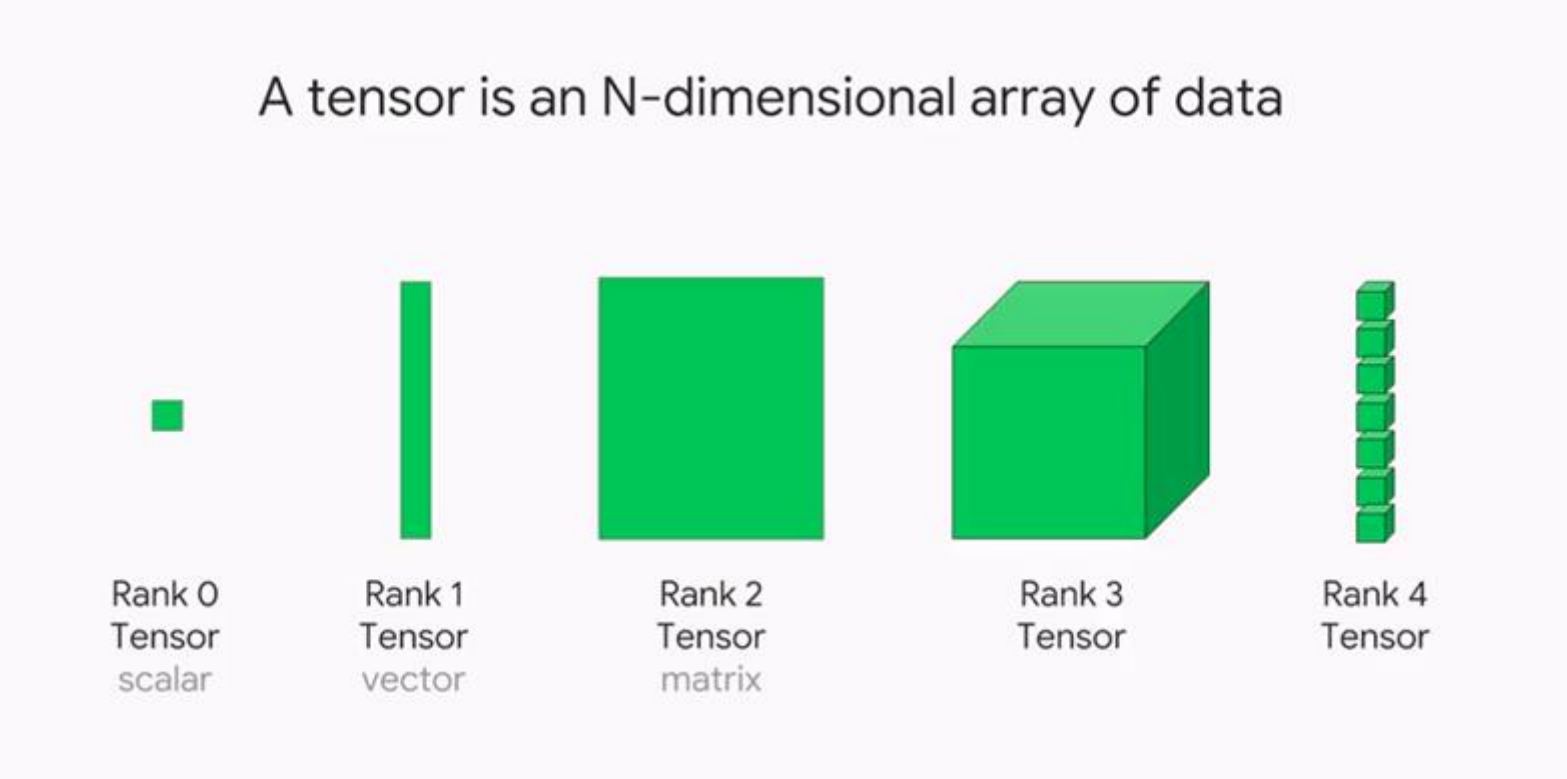
How to make a modern DL framework



How to make a modern DL framework

Tensors

Abstraction to higher order of data



How to make a modern DL framework

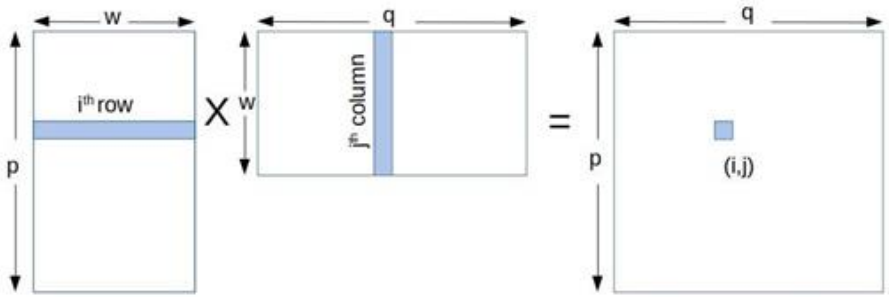
CPU

GPU

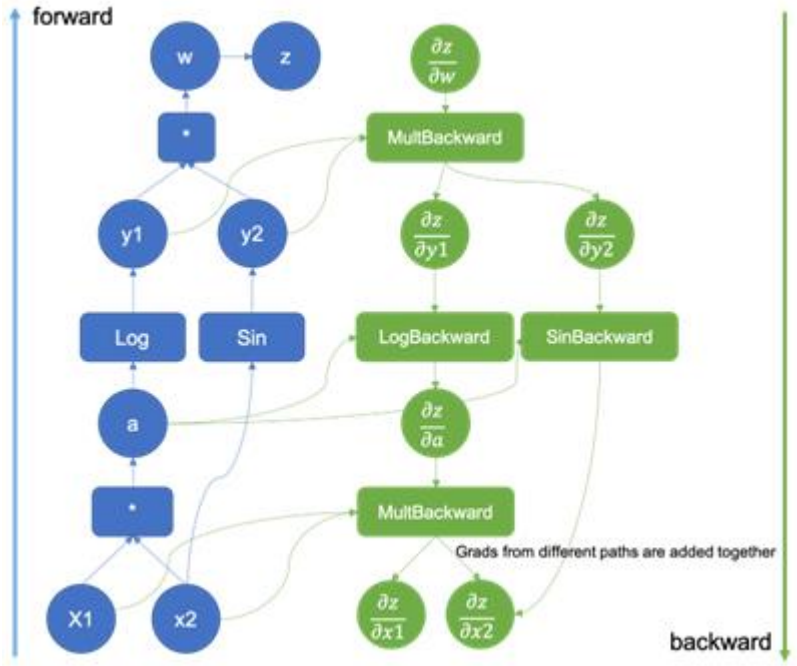
TPU

Hardware acceleration

Faster computations



How to make a modern DL framework



Automatic differentiation

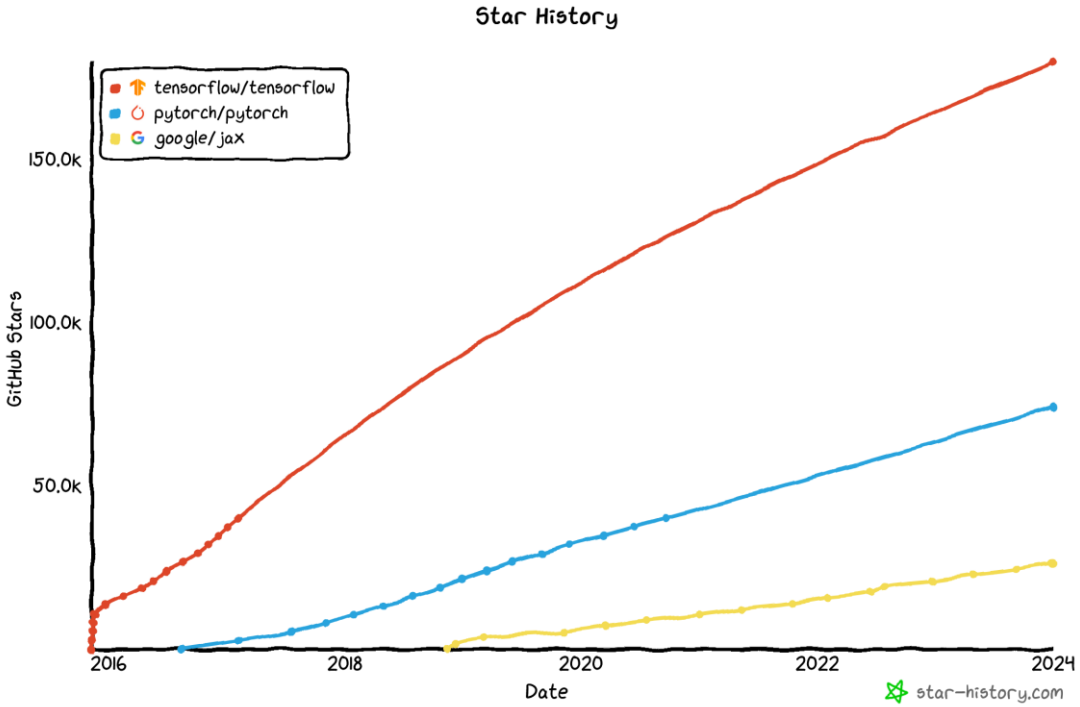
Ease of use

```
(base) C:\Users\nsde>python
Python 3.8.5 (default, Sep  3 2020, 21:29:08) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import torch
>>> 2*torch.ones(5, requires_grad=True)
tensor([2., 2., 2., 2., 2.], grad_fn=<MulBackward0>)
>>>
```

The current landscape

Pytorch vs Tensorflow vs Jax all support the same underlying feature set

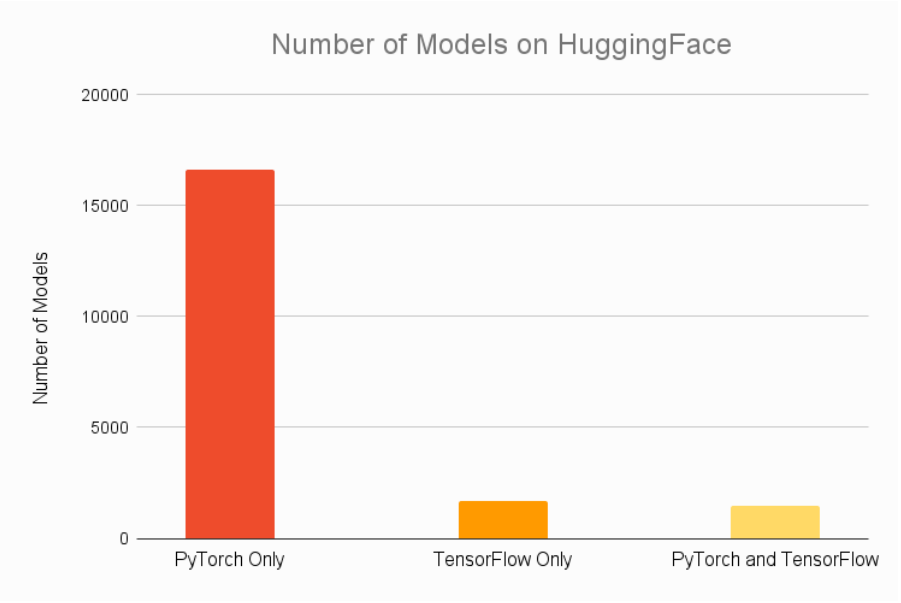
- Easy to use Python interface
- Hardware acceleration
- Research and industry specific features



The current landscape

In this course we use Pytorch because

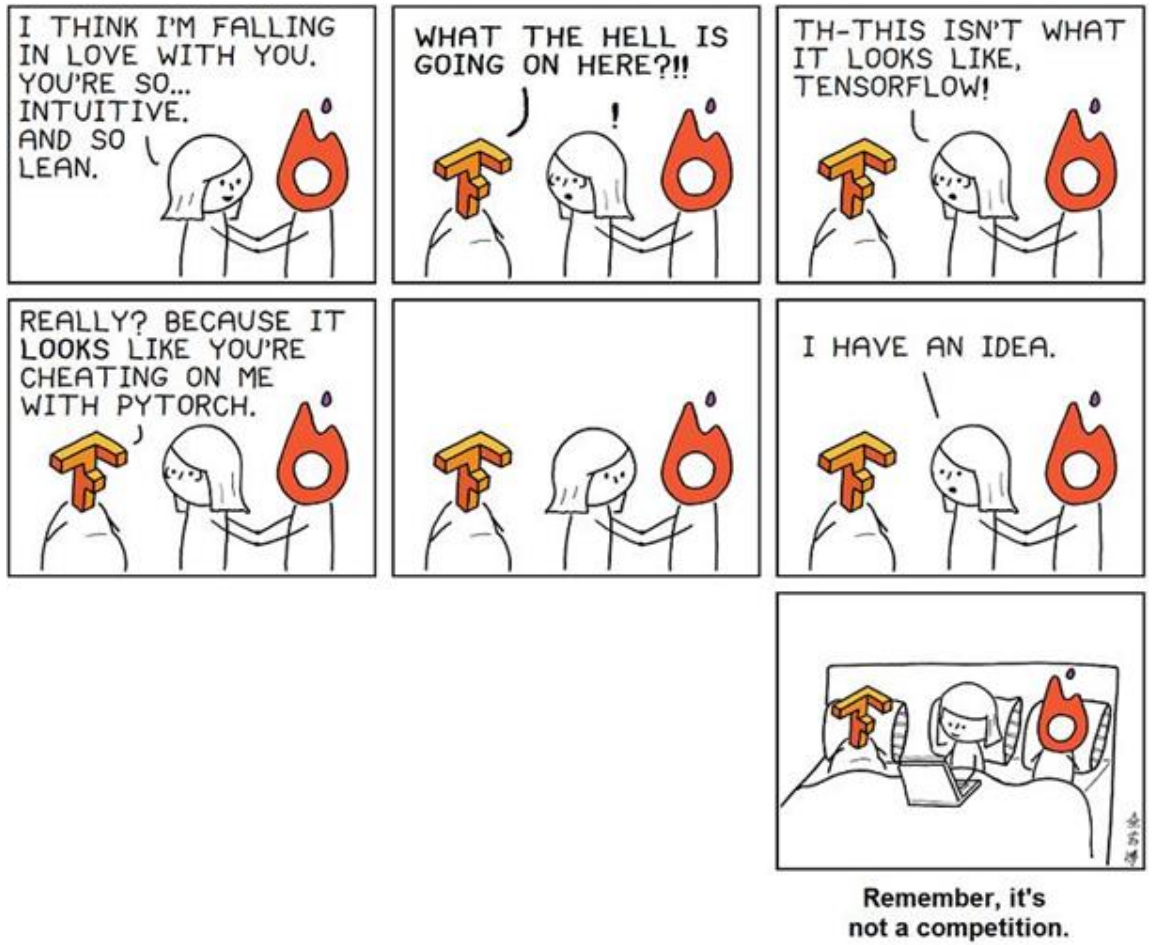
- 💡 Absolutely dominant framework (#models, #papers, #competition winners etc.)
- 💡 What we use locally for research



Reference

I highly recommend…

If you have the time, learn the basics of them all 😳



In practice, people often use high-level frameworks

Makes a lot of coding much easier.

Recommend, to only use these if you understand the underlying framework.

We get back top one of these.

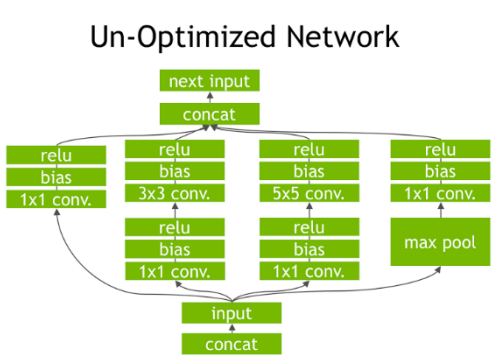
Haiku

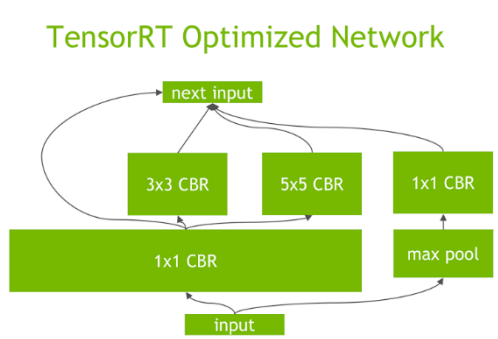
Sonnet

Current trends in deep learning software

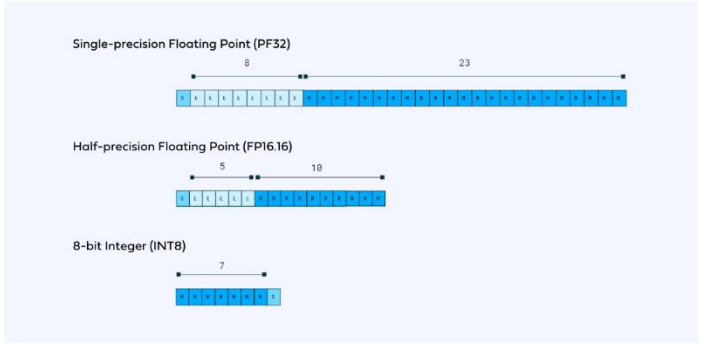
- More accelerators

- Compiled models





- Lower precision computations



Meme of the day