

Introduction

M. Soleymani

Deep Learning

Sharif University of Technology

Spring 2026

Course Info

- Course Number: 40-719 (Time: Sat-Mon 15:00-16:30 Location: CE 201)
- Instructor: Mahdieh Soleymani (soleymani@sharif.edu)
- Head TA: **Hossein Shahabadi** (@webixen)
- Office hours: Sat-Mon 10:30-12:00
 - set appointment through email

Communication

- Quera
 - Policies and rules
 - Tentative schedule
 - Slides and notes
 - Assignments
 - Discussions
 - Ask questions about homework, grading, logistics
 - communication with course staff
- Email
 - Private questions

Materials

- (Optional) text books
 - **Deep Learning**, by Ian Goodfellow, Yoshua Bengio, and Aaron Courville, 2016. ([free version](#))
 - **Understanding Deep Learning**, by Simon J.D. Prince, 2023. ([free draft version](#))
 - **Deep Learning: Foundations and Concepts**, by C. Bishop and H. Bishop, 2023.
 - **Dive into deep learning**, by Zhang et al., 2023 ([free version](#))
- Some papers
- Notes, lectures, and demos

Marking Scheme

- | | |
|-------------------------|-----|
| • Midterm Exam: | 20% |
| • Final Exam: | 30% |
| • Quiz: | 10% |
| • Homework assignments: | 40% |

About homework assignments

- 5 assignments (both theoretical and practical)
- HWs are implementation-heavy
 - A lot of coding and experimenting
 - In some assignments, you deal with large datasets
- Language of choice: Python
- Toolkit of choice: Pytorch

Assignments: Late policy

- Everyone gets up to some slack days for assignments (but there is also constraint for slack days of each assignment)
- You can distribute them across your assignments
- Once you use up your slack days, all subsequent late submissions will accrue penalty according to the policy specified in the Rules file.

Collaboration policy

- We follow the [CE Department Honor Code](#) – read it carefully.
- Don't look at solutions or code of others; everything you submit should be your own work
- Don't share your solution or code with others although discussing general ideas is fine and encouraged
- Indicate in your submissions anyone you worked with or any tool you used

Prerequisites

- Machine Learning
- Knowledge of calculus and linear algebra
 - You can review it from this book: [Mathematics of deep learning](#)
- Programming (Python)
- Time and patience

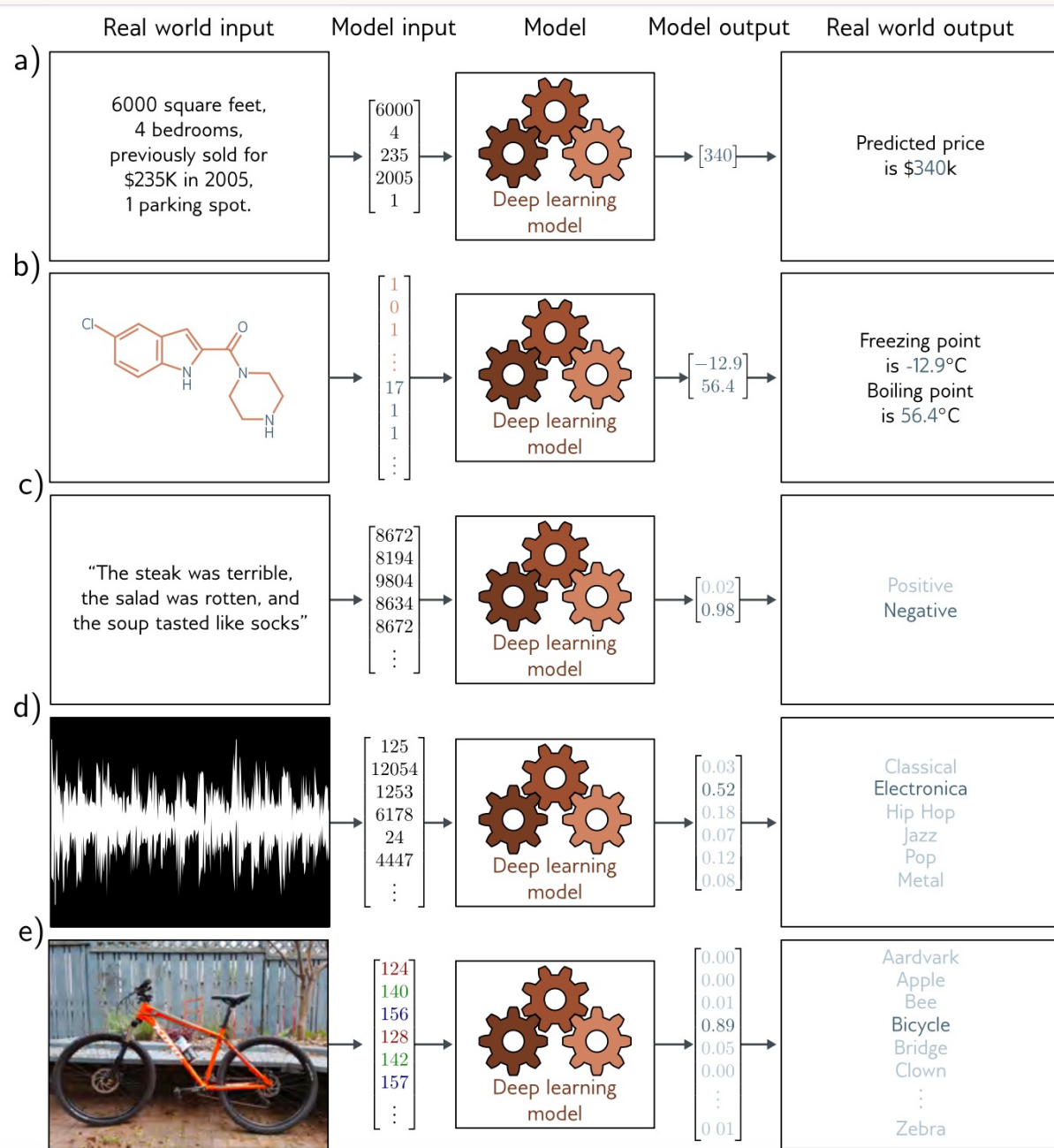
TA Section

- Hands-on tutorials, with more practical details than the main lecture
- Workshops on some topics

Overview

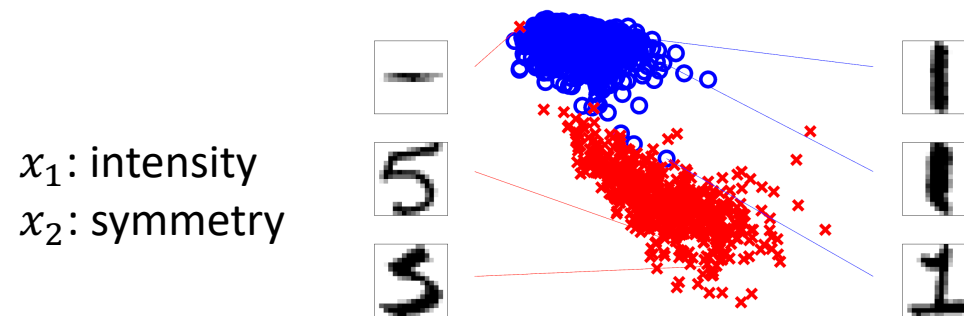
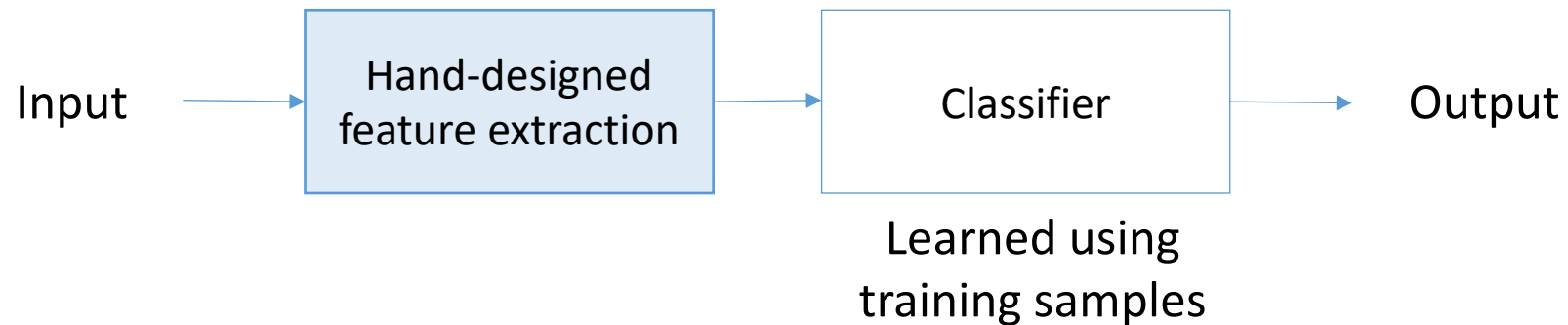
Deep learning

- Learning a computational models consists of multiple processing layers
 - learn representations of data with multiple levels of abstraction.
- Dramatically improved the state-of-the-art in many tasks



Machine Learning Methods

- Conventional machine learning methods:
 - try to learn the mapping from the input features to the output by samples
 - However, they need appropriately designed hand-designed features



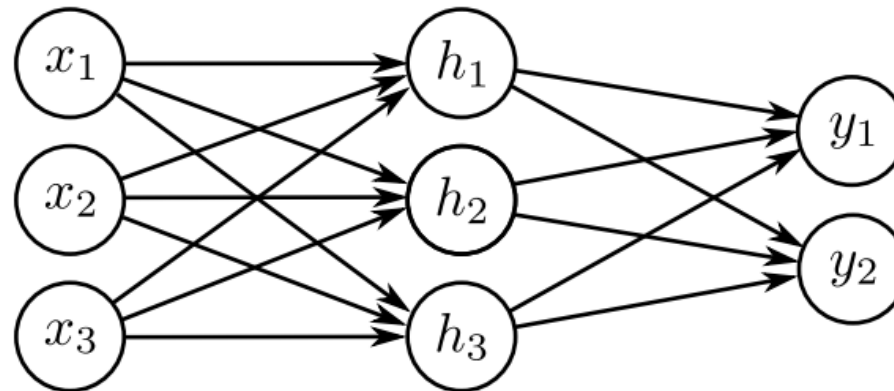
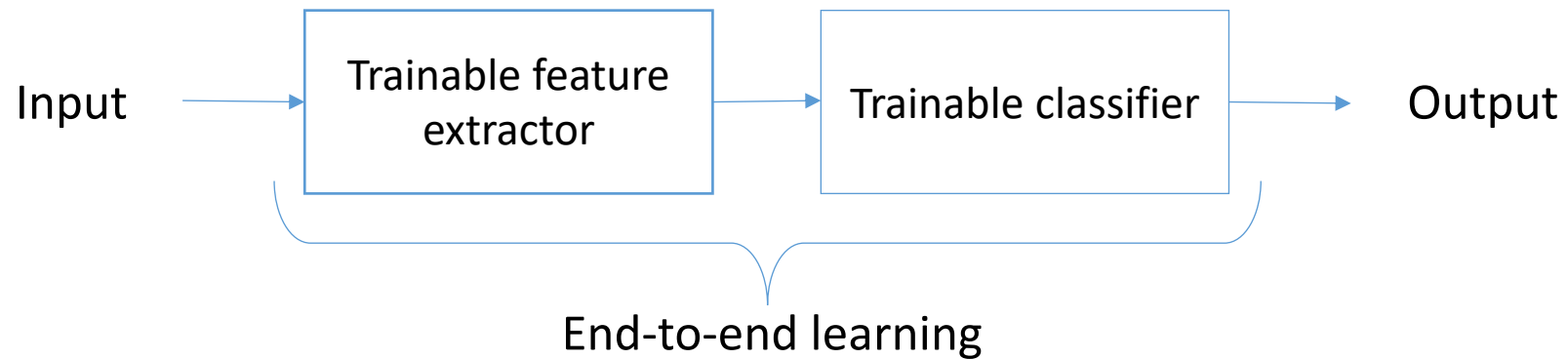
[Abu Mostafa, 2012]

Representation of Data

- Traditional learning methods depends heavily on the data representation
 - **Most efforts were on designing proper features**
- However, designing hand-crafted features for inputs like image, videos, time series, and sequences is not trivial at all.
 - It is difficult to know which features should be extracted.
 - Sometimes, it needs long time for a community of experts to find (an incomplete and over-specified) set of these features.

Representation Learning

- Using learning to discover both:
 - the representation of data from input features
 - and the mapping from representation to output

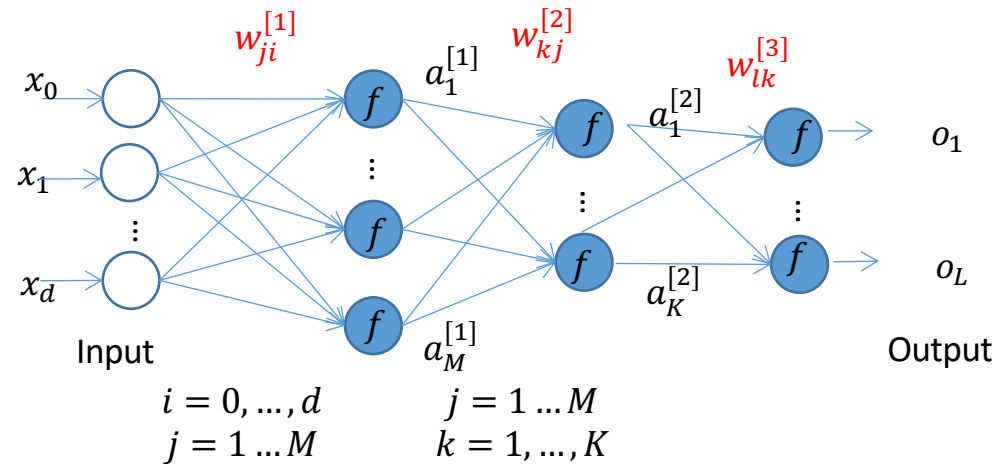


Previous Representation Learning Methods

- Metric learning and kernel learning methods attempted to solve this problem
 - However, they were shallow models for feature (or representation) learning
- DL finds representations that are expressed in terms of other, simpler representations
 - Usually hierarchical representation is meaningful and useful

Multi-layer Neural Network

- A multilayer perceptron is just a mapping input values to output values.
 - The function is formed by composing many simpler functions.
 - These middle layers are not given in the training data must be determined

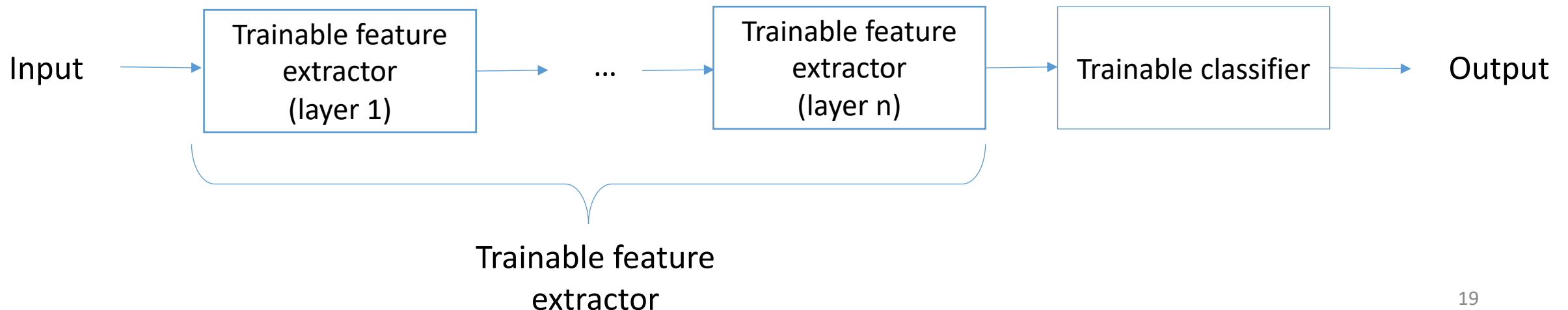


Example of f functions:
 $f(z) = \max(0, z)$

$$a_k^{[l]} = f\left(\sum_{i=0}^M w_{ki}^{[l]} a_i^{[l-1]}\right)$$

Deep Learning Approach

- Deep breaks the desired complicated mapping into a series of nested simple mappings
 - each mapping described by a layer of the model.
 - each layer extracts features from output of previous layer
- shows impressive performance on many Artificial Intelligence tasks



Deep Representations: The Power of Compositionality

- Deep learning has great power and flexibility by learning to represent the world as a nested hierarchy of concepts
- Compositionality is useful to describe the world around us efficiently
 - Learned function seen as a composition of simpler operations
 - Hierarchy of features, concepts, leading to more abstract factors enabling better generalization
 - each concept defined in relation to simpler concepts
 - more abstract representations computed in terms of less abstract ones.
 - Again, theory shows this can be exponentially advantageous

Example of Nested Representation

Faces



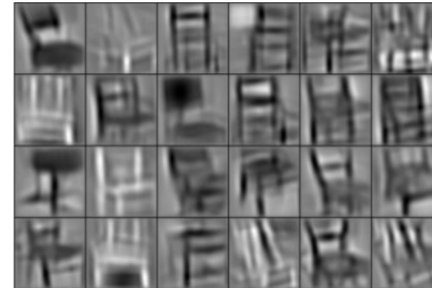
Cars



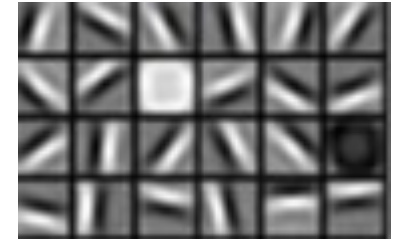
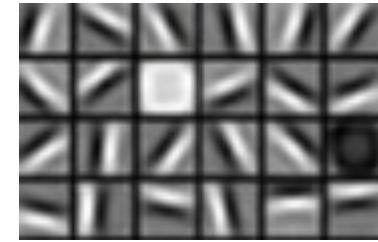
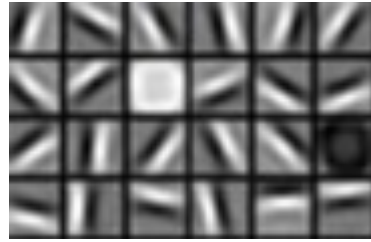
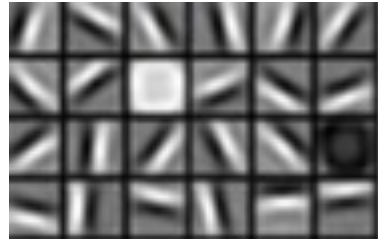
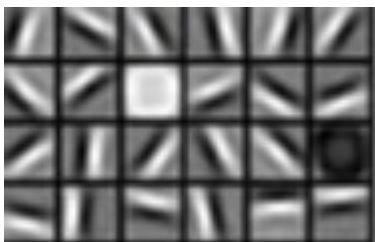
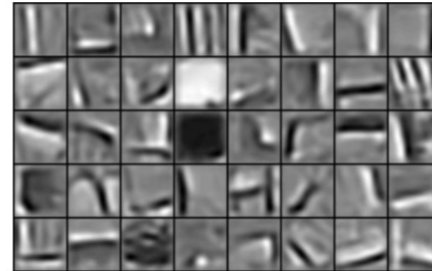
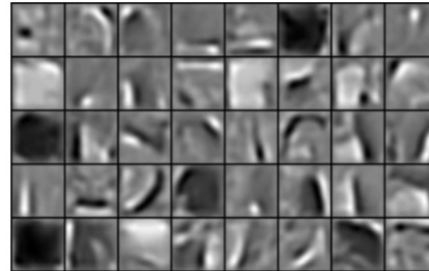
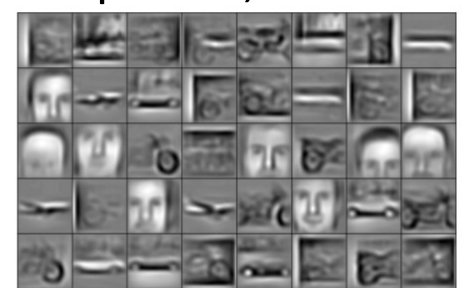
Elephants



Chairs



Faces, Cars,
Elephants, and Chairs



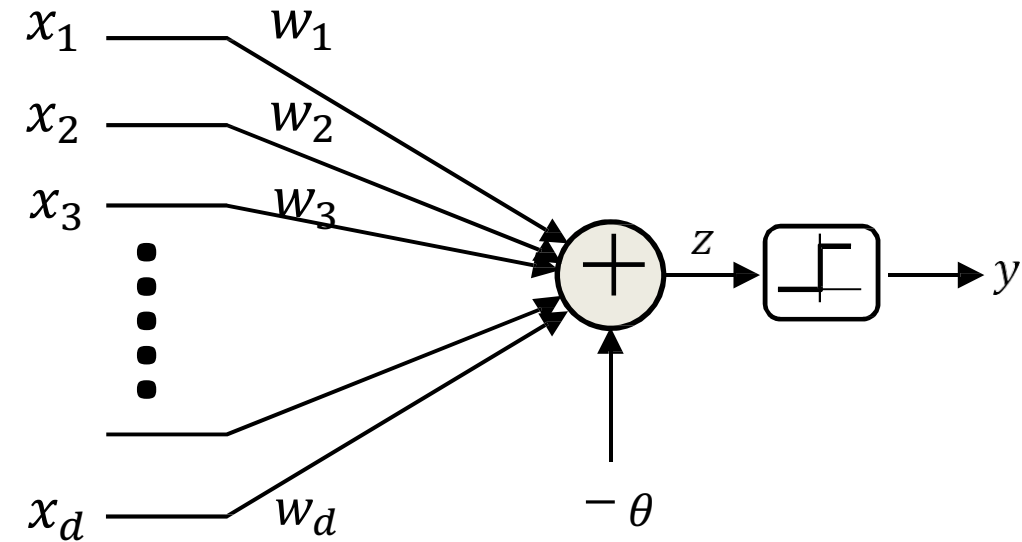
Lee et al., ICML 2009

Deep Learning Brief History

- 1940s–1960s:
 - development of theories of biological learning
 - implementations of the first models
 - perceptron (Rosenblatt, 1958) for training of a single neuron.
- 1980s-1990s: back-propagation algorithm to train a neural network with more than one hidden layer
 - too computationally costly to allow much experimentation with the hardware available at the time.
 - Small datasets
- 2006 “Deep learning” name was selected
 - ability to train deeper neural networks than had been possible before
 - Although began by using unsupervised representation learning, later success obtained usually using large datasets of labeled samples

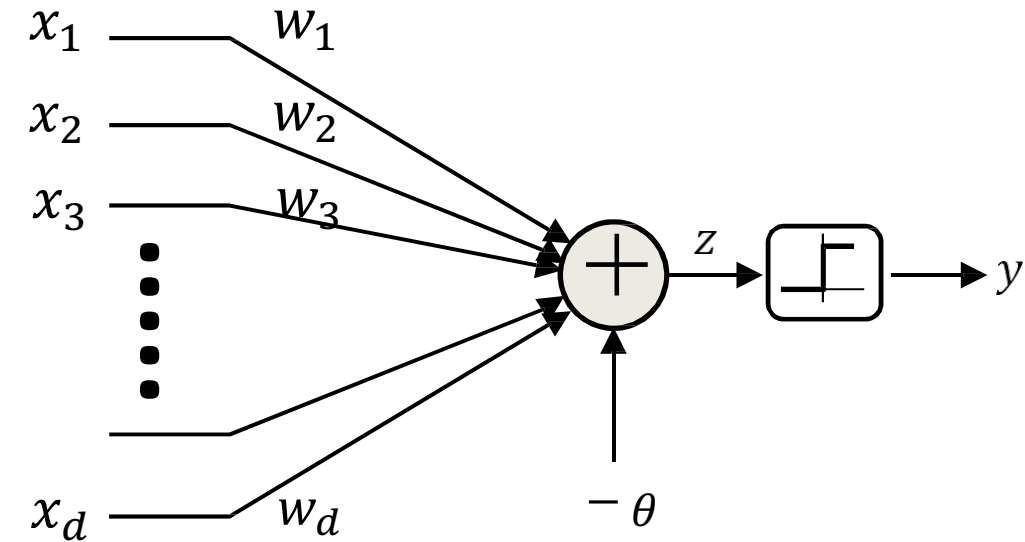
Deep Learning History: Timeline

- **1943 Artificial Neuron**
- 1957 Perceptron
- 1969 Limitations of Neural Networks
- 1979 Neocognitron (inspires CNNs)
- 1982 Recurrent Neural Networks (RNNs)
- 1986 Back propagation
- 1997 LSTM
- 1998 LeNet (Neocognitron+Backprop)
- 2006 Deep Learning
- 2009 ImageNet
- 2012 AlexNet



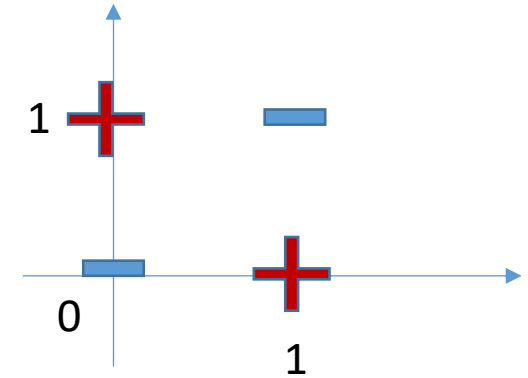
Deep Learning History: Timeline

- 1943 Artificial Neuron
- **1957 Perceptron**
- 1969 Limitations of Neural Networks
- 1979 Neocognitron (inspires CNNs)
- 1982 Recurrent Neural Networks (RNNs)
- 1986 Back propagation
- 1997 LSTM
- 1998 LeNet (Neocognitron+Backprop)
- 2006 Deep Learning
- 2009 ImageNet
- 2012 AlexNet



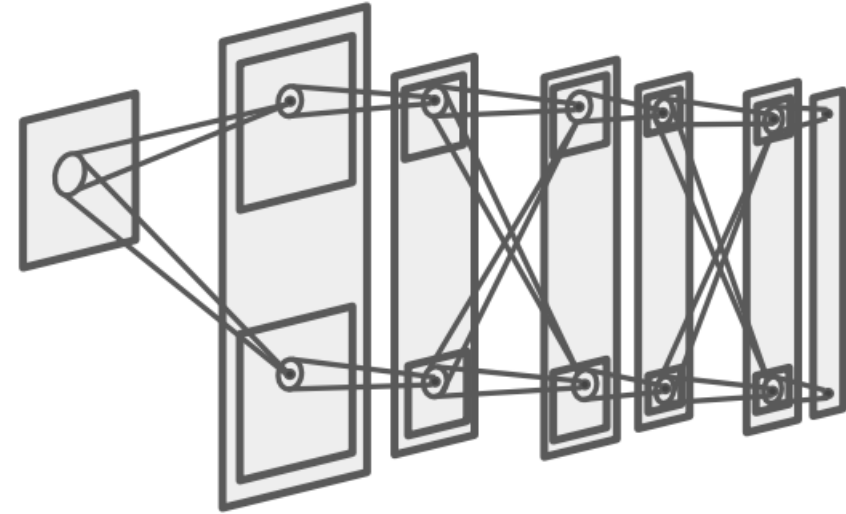
Deep Learning History: Timeline

- 1943 Artificial Neuron
- 1957 Perceptron
- **1969 Limitations of Neural Networks**
- 1979 Neocognitron (inspires CNNs)
- 1982 Recurrent Neural Networks (RNNs)
- 1986 Back propagation
- 1997 LSTM
- 1998 LeNet (Neocognitron+Backprop)
- 2006 Deep Learning
- 2009 ImageNet
- 2012 AlexNet



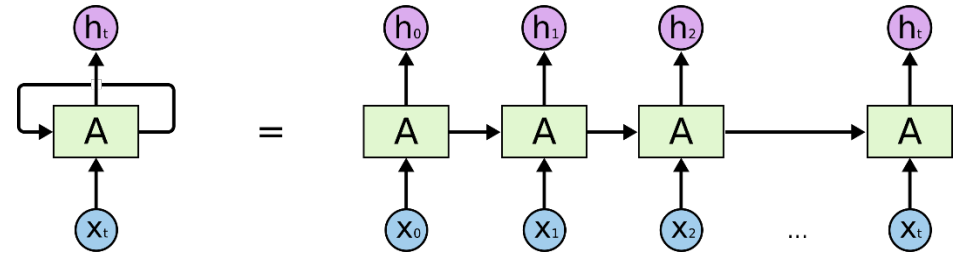
Deep Learning History: Timeline

- 1943 Artificial Neuron
- 1957 Perceptron
- 1969 Limitations of Neural Networks
- **1979 Neocognitron (inspires CNNs)**
- 1982 Recurrent Neural Networks (RNNs)
- 1986 Back propagation
- 1997 LSTM
- 1998 LeNet (Neocognitron+Backprop)
- 2006 Deep Learning
- 2009 ImageNet
- 2012 AlexNet



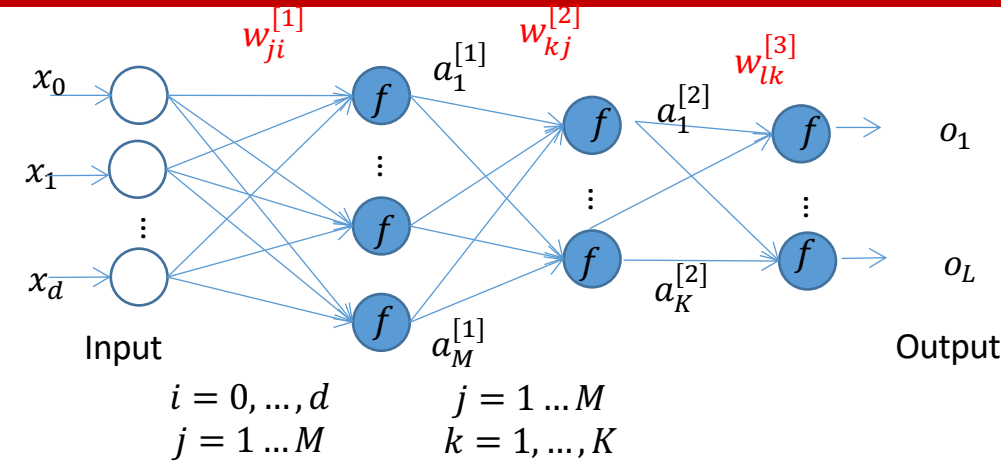
Deep Learning History: Timeline

- 1943 Artificial Neuron
- 1957 Perceptron
- 1969 Limitations of Neural Networks
- 1979 Neocognitron (inspires CNNs)
- **1982 Recurrent Neural Networks (RNNs)**
- 1986 Back propagation
- 1997 LSTM
- 1998 LeNet (Neocognitron+Backprop)
- 2006 Deep Learning
- 2009 ImageNet
- 2012 AlexNet



Deep Learning History: Timeline

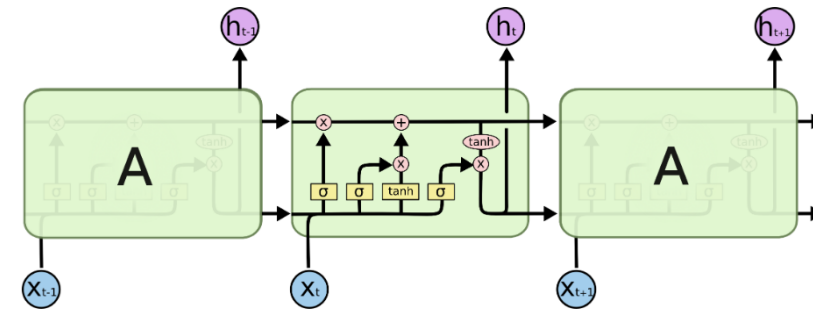
- 1943 Artificial Neuron
- 1957 Perceptron
- 1969 Limitations of Neural Networks
- 1979 Neocognitron (inspires CNNs)
- 1982 Recurrent Neural Networks (RNNs)
- **1986 Back propagation** (previously invented as automatic differentiation in 1970)
- 1997 LSTM
- 1998 LeNet (Neocognitron+Backprop)
- 2006 Deep Learning
- 2009 ImageNet
- 2012 AlexNet



$$\frac{\partial E}{\partial w_{kj}^{[l]}} = \frac{\partial E}{\partial a_k^{[l]}} \frac{\partial a_k^{[l]}}{\partial w_{kj}^{[l]}}$$
$$\frac{\partial E}{\partial a^{[l]}} = \frac{\partial a^{[l+1]}}{\partial a_l^{[l]}} \frac{\partial E}{\partial a^{[l+1]}}$$

Deep Learning History: Timeline

- 1943 Artificial Neuron
- 1957 Perceptron
- 1969 Limitations of Neural Networks
- 1979 Neocognitron (inspires CNNs)
- 1982 Recurrent Neural Networks (RNNs)
- 1986 Back propagation
- **1997 LSTM**
- 1998 LeNet (Neocognitron+Backprop)
- 2006 Deep Learning
- 2009 ImageNet
- 2012 AlexNet

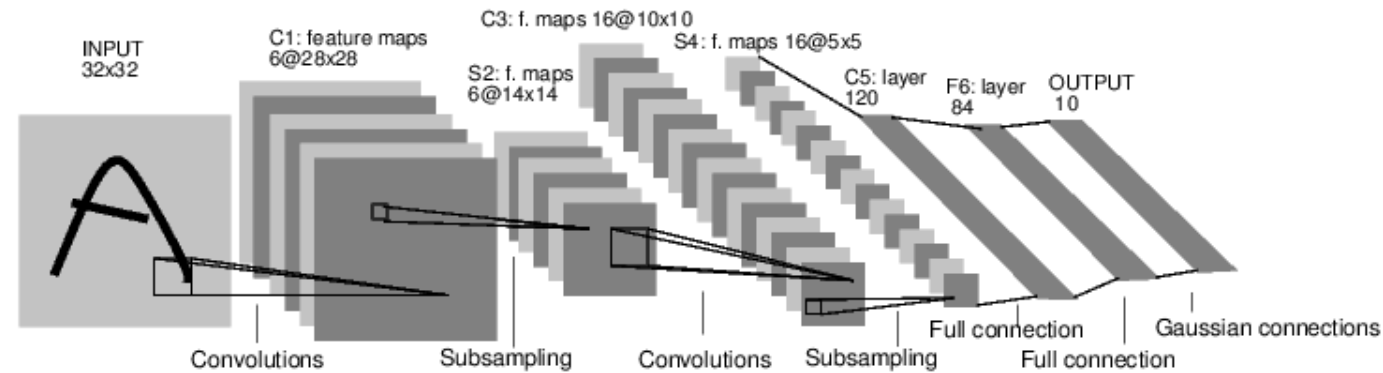


Source: Colah's blog

Deep Learning History: Timeline

- 1943 Artificial Neuron
- 1957 Perceptron
- 1969 Limitations of Neural Networks
- 1979 Neocognitron (inspires CNNs)
- 1982 Recurrent Neural Networks (RNNs)
- 1986 Back propagation
- 1997 LSTM
- **1998 LeNet (Neocognitron+Backprop)**
- 2006 Deep Learning
- 2009 ImageNet
- 2012 AlexNet

LeNet: Handwritten Digit Recognition (recognizes zip codes)
Training Sample : 9298 zip codes on mails



[LeNet, Yann Lecun, et. al, 1989]

Deep Learning History: Timeline

- 1943 Artificial Neuron
- 1957 Perceptron
- 1969 Limitations of Neural Networks
- 1979 Neocognitron (inspires CNNs)
- 1982 Recurrent Neural Networks (RNNs)
- 1986 Back propagation
- 1997 LSTM
- 1998 LeNet (Neocognitron+Backprop)
- **2006 Deep Learning**
- 2009 ImageNet
- 2012 AlexNet

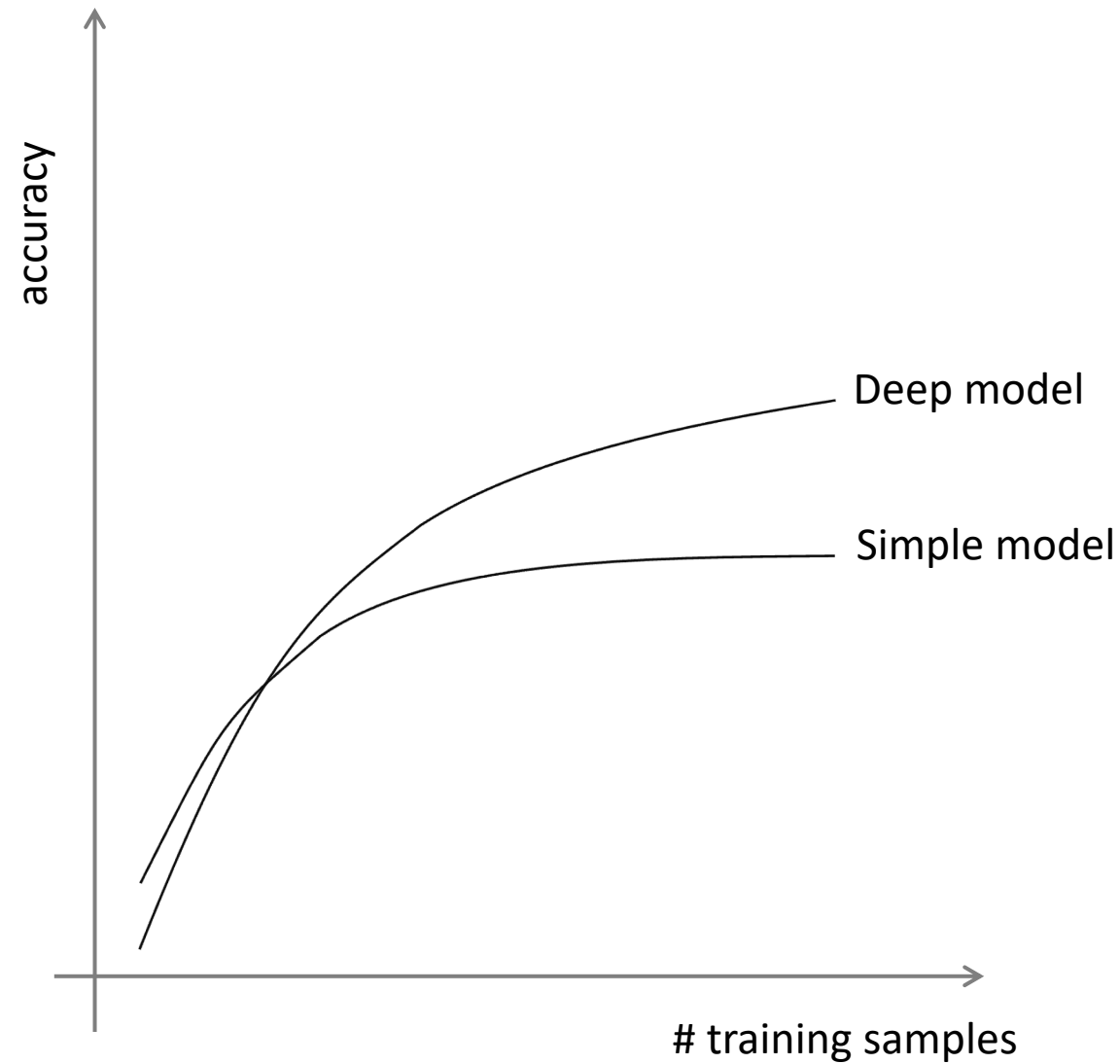
The training of each layer individually is an easier undertaking

- Training multi layered neural networks became easier
- Per-layer trained parameters initialize further training

[Deng, Dong, Socher, Li, Li, & Fei-Fei, 2009]

- 22K categories and 14M images
 - Collected from web & labeled by Amazon Mechanical Turk
- The Image Classification Challenge:
 - Imagenet Large Scale Visual Recognition Challenge (ILSVRC)
 - 1,000 object classes
 - 1,431,167 images
- Much larger than the previous datasets of image classification

Large Datasets



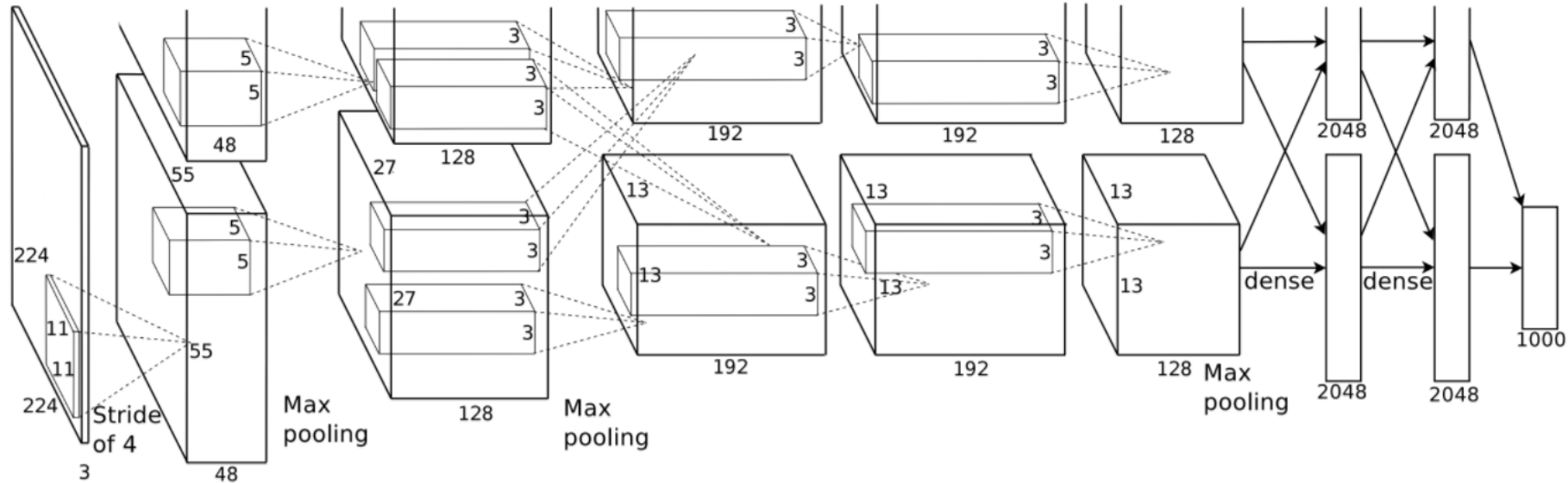
Why does deep learning become popular?

- Data: Large datasets
- Hardware: Availability of the computational resources to run much larger models
- Algorithm
 - New training techniques
 - New models
 - Frameworks

IMGENET



Alexnet (2012)

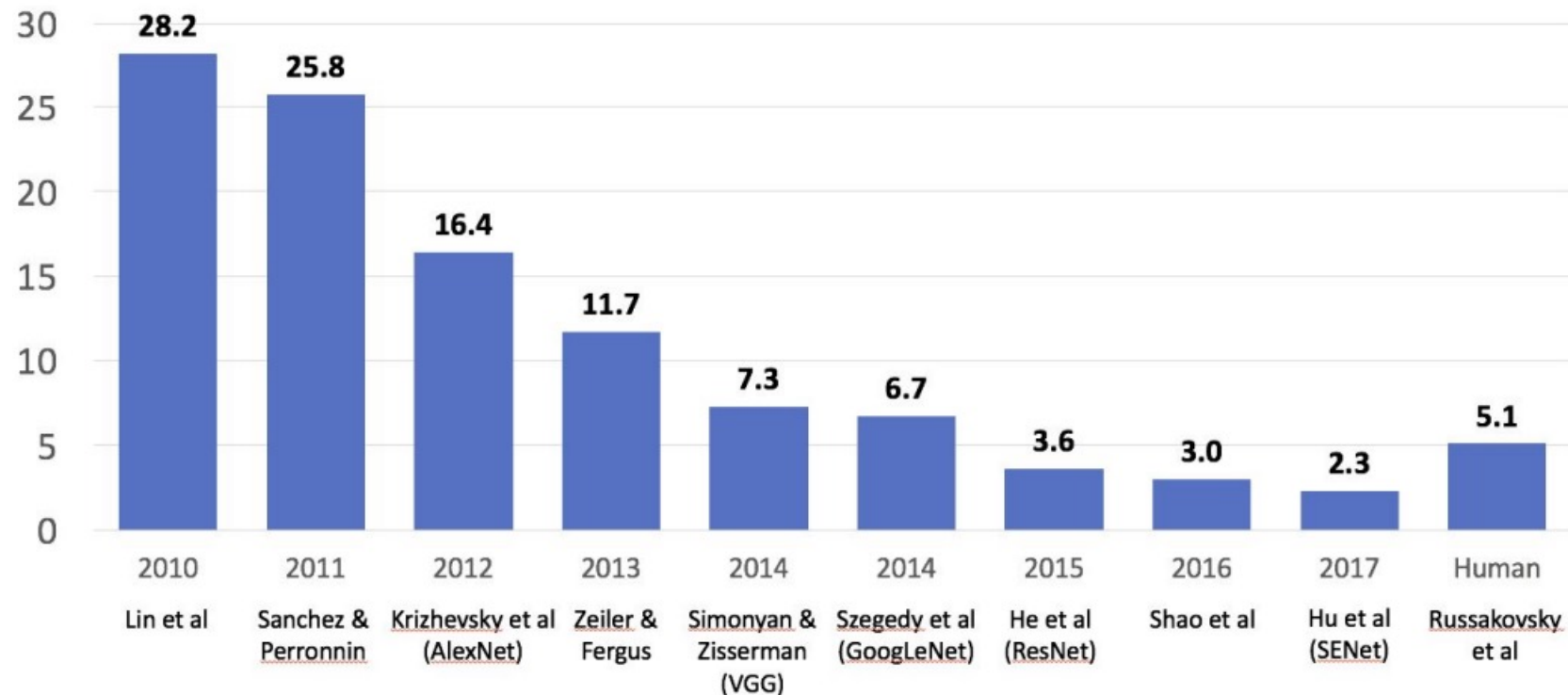


Krizhevsky, Alex, Sutskever, and Hinton, Imagenet classification with deep convolutional neural networks, NIPS 2012

- Reduces 25.8% top 5 error of the winner of 2011 challenge to 16.4%

Deep Models for Image Classification (ILSVRC)

- 5.1% is the performance of human on this data set



Using Pre-trained Models

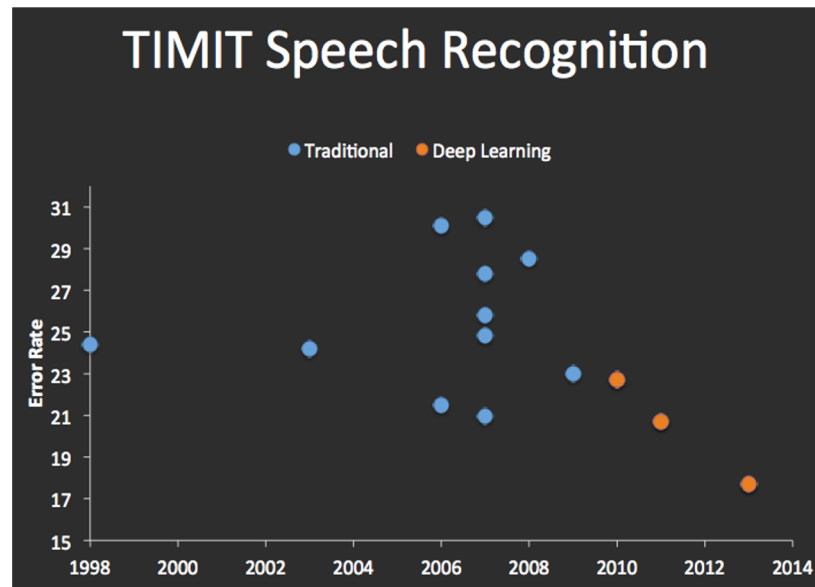
- We don't have large-scale datasets on all image tasks and also we may not have time to train such deep networks from scratch
- On the other hand, learned weights for popular networks (on ImageNet) are available.
- Use pre-trained weights of these networks (other than final layers) as generic feature extractors for images
- Works better than handcrafted feature extraction on natural images

Other Vision Tasks

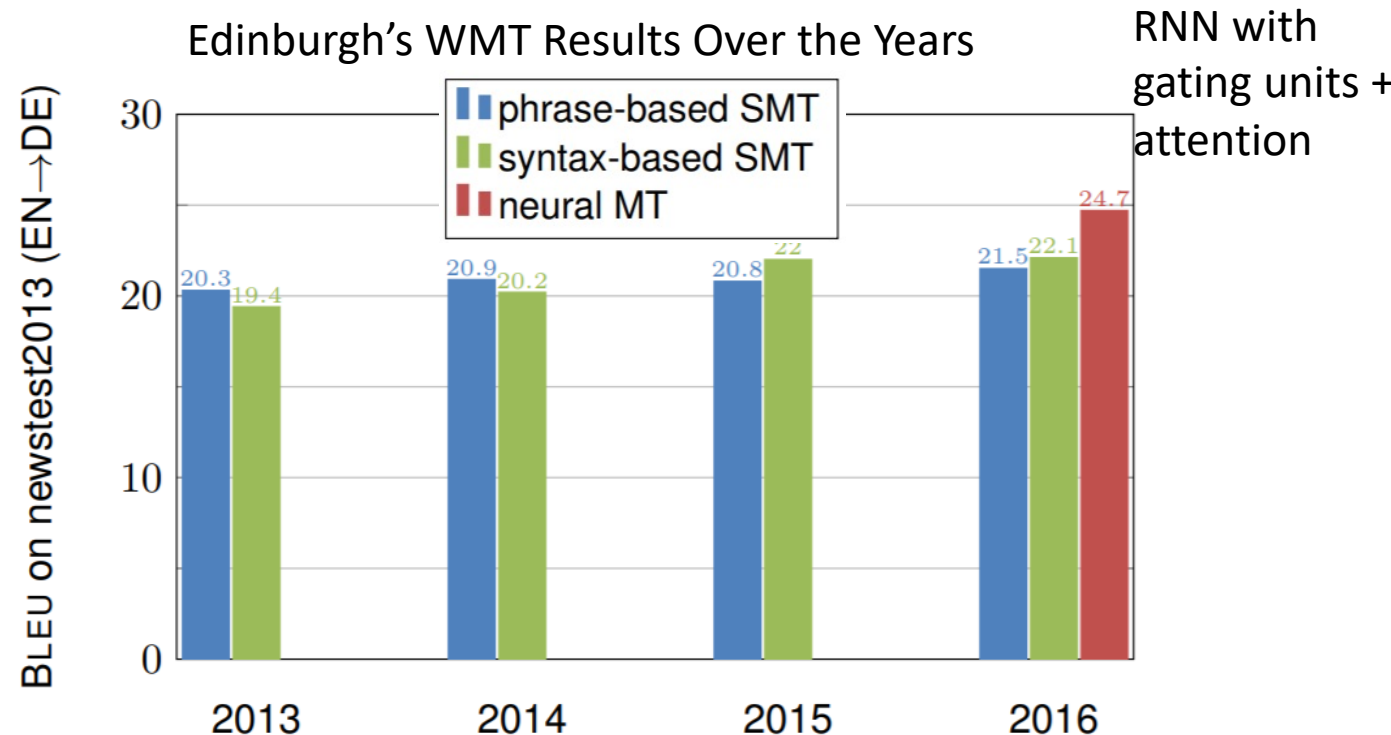
- After image classification, achievements were obtained in other vision tasks:
 - Object detection
 - Segmentation
 - Image captioning
 - Visual Question Answering (VQA)
 - ...

Similar Trends also in Speech and NLP

- Deep learning became SOTA also in speech and NLP tasks



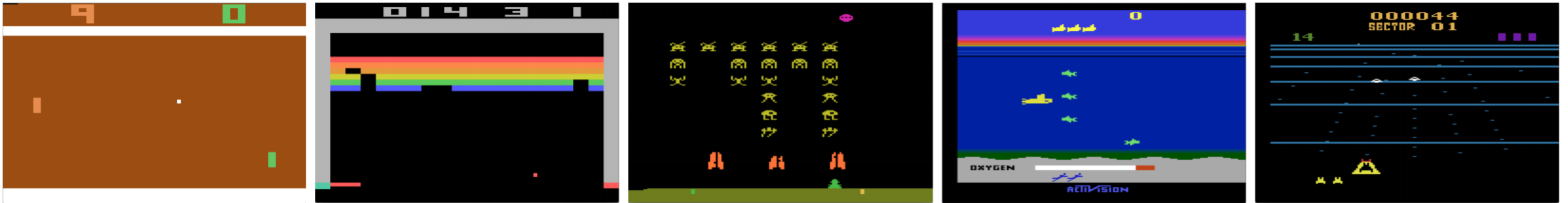
Source: clarifai



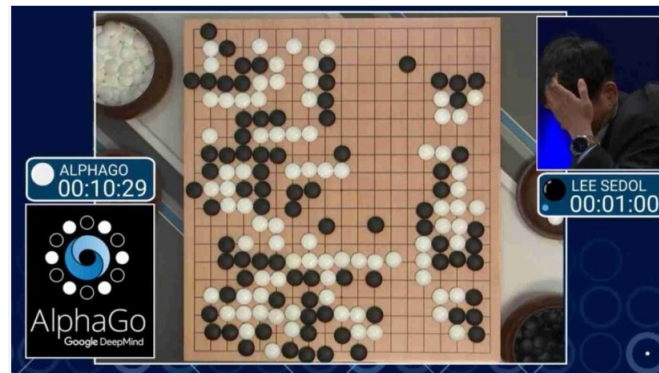
Source: http://www.meta-net.eu/events/meta-forum2016/slides/09_sennrich.pdf

Games

- DQN (2013): Atari 2600 games
 - neural network agent that is able to successfully learn to play as many of the games as possible without any hand-designed feature.

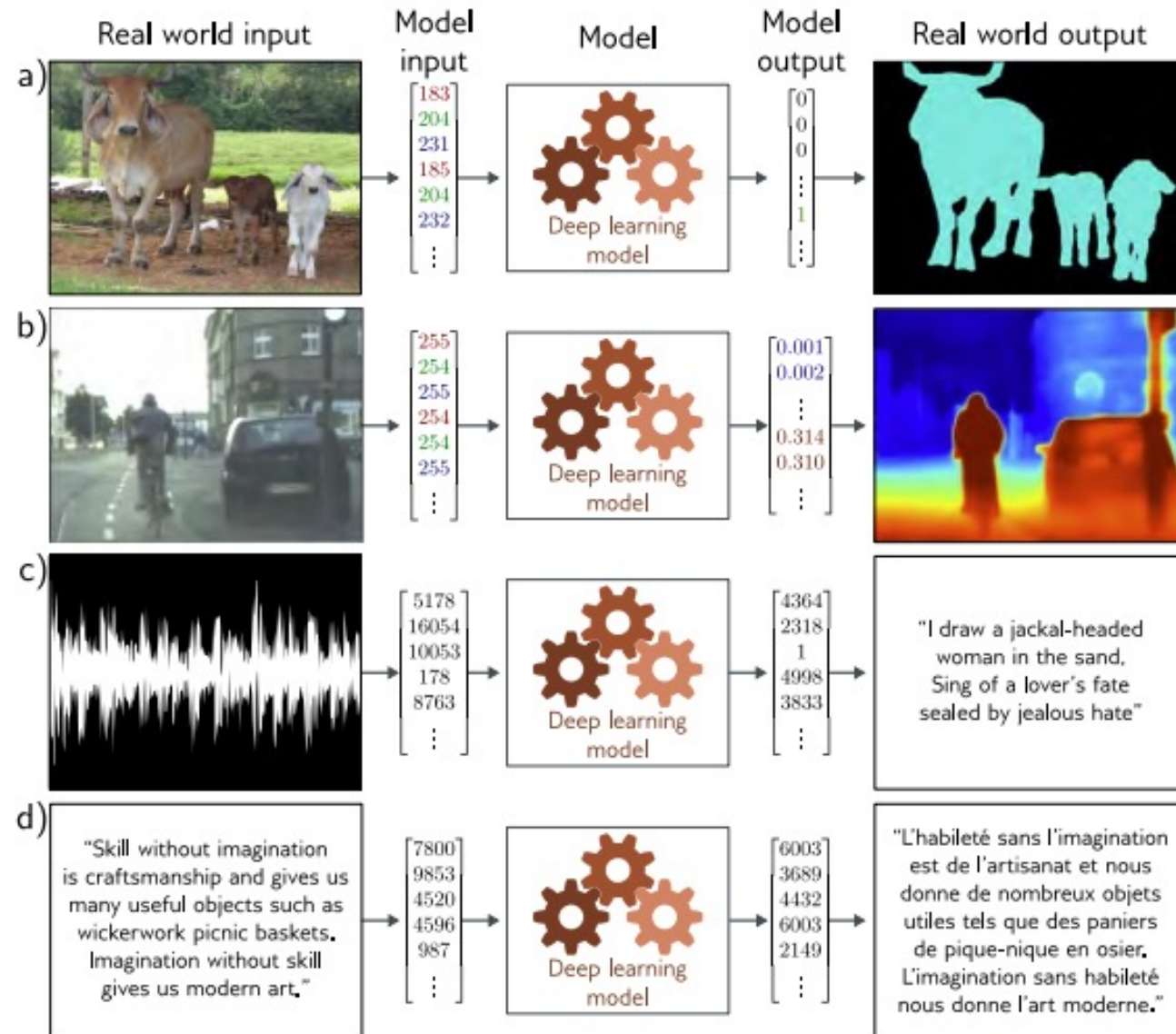


- Deep Mind's alphaGo defeats former world champion in 2016.



Source: <https://gogameguru.com/alphago-shows-true-strength-3rd-victory-lee-sedol/>

Some Examples



Deep Learning is Everywhere

- Vision
- NLP
- Speech
- Games
- Bioinformatics
- ...

nature View all Nature Research

[Explore content](#) [Journal information](#) [Publish with us](#) [Subscribe](#)

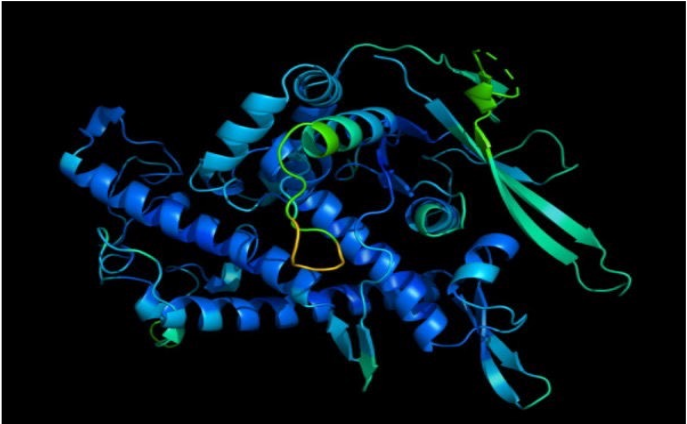
[nature](#) > [news](#) > [article](#)

NEWS · 30 NOVEMBER 2020

‘It will change everything’: DeepMind’s AI makes gigantic leap in solving protein structures

Google’s deep-learning program for determining the 3D shapes of proteins stands to transform biology, say scientists.

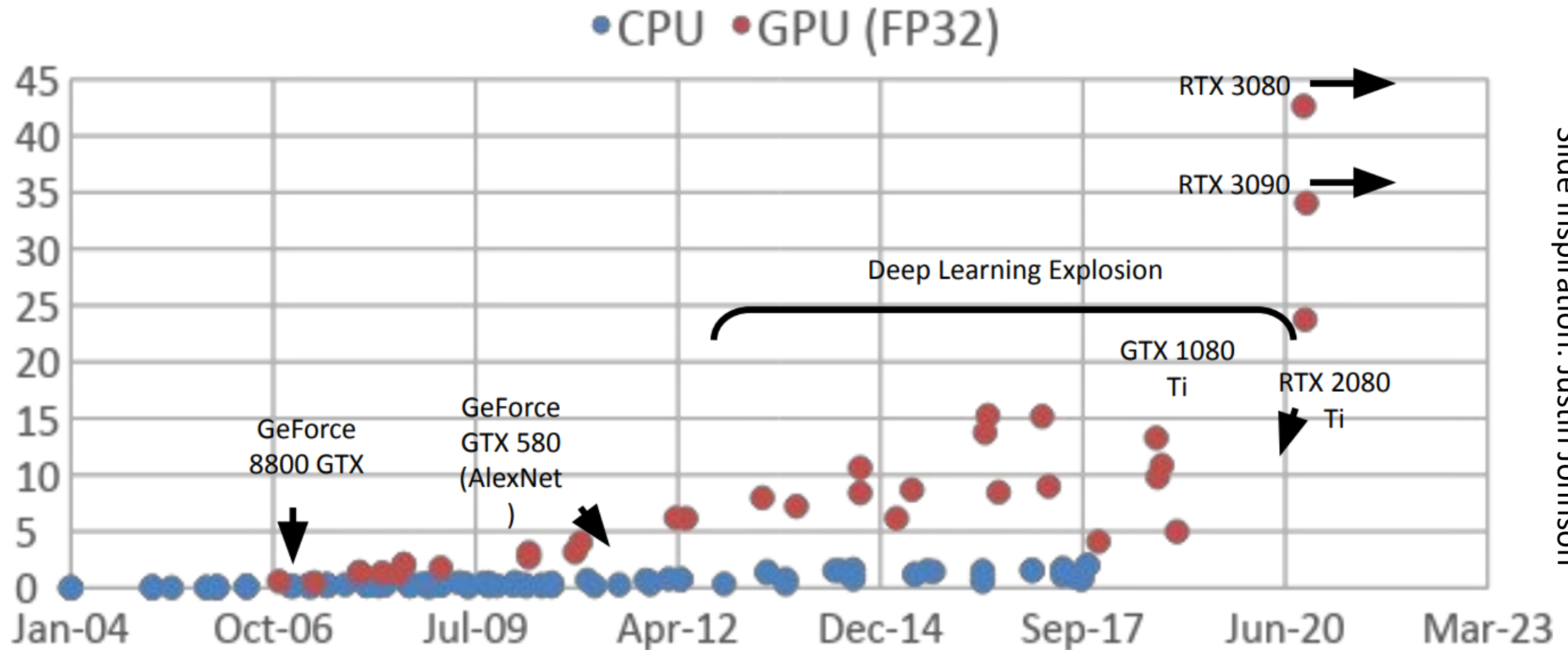
[Ewen Callaway](#)



A protein's function is determined by its 3D shape. Credit: DeepMind

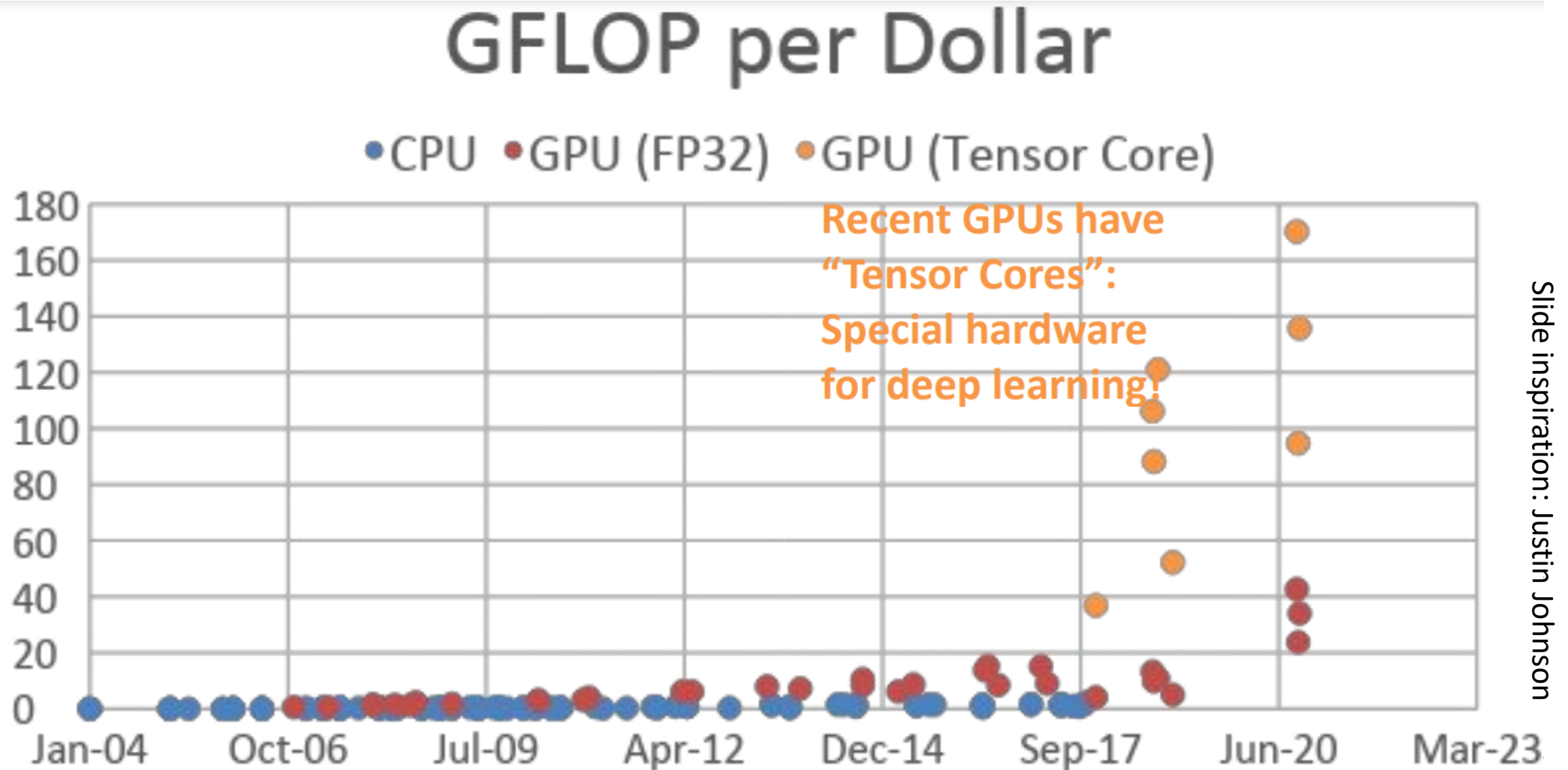
Computational Resources

GFLOP per Dollar



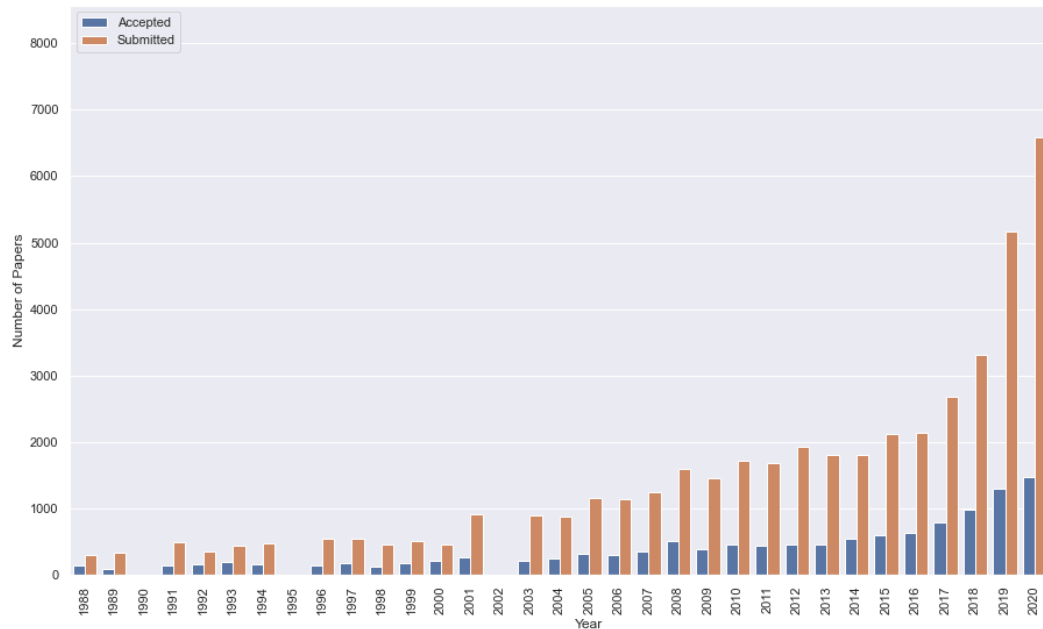
Slide inspiration: Justin Johnson

Computational Resources

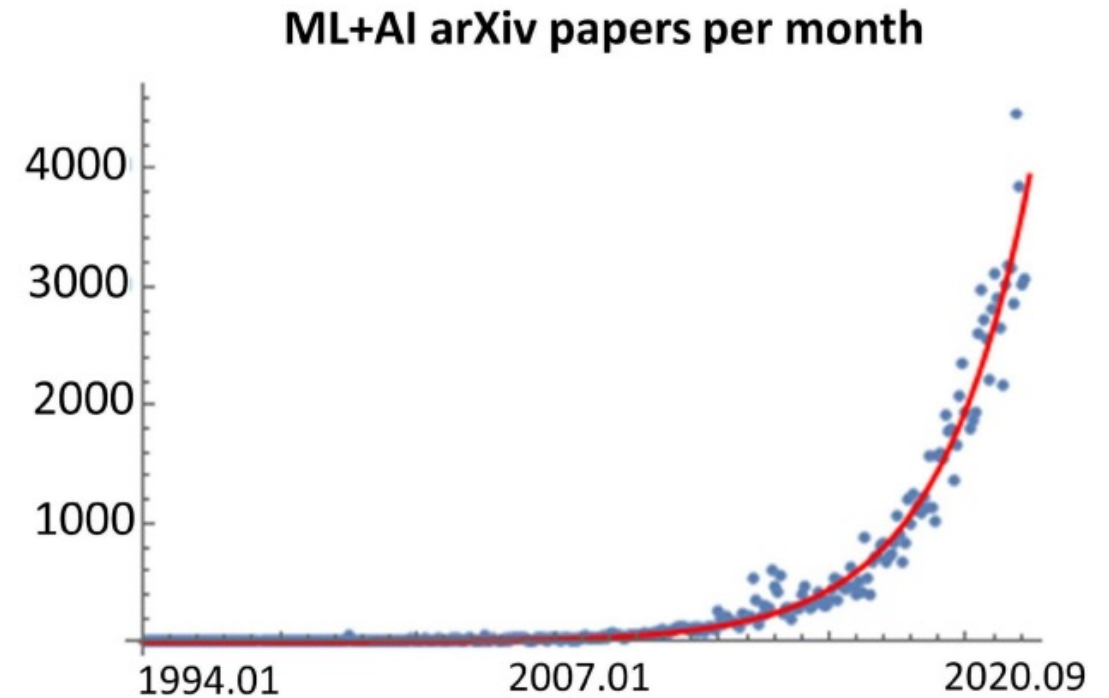


Slide inspiration: Justin Johnson

2012 to Present: Deep Learning Explosion



CVPR papers

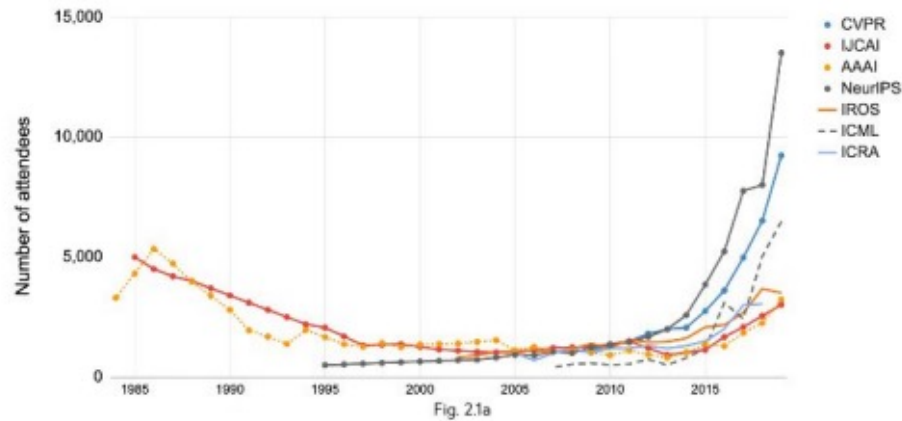


ML+AI papers

Source:
<https://twitter.com/MarioKrenn6240/status/1314622995139264517>

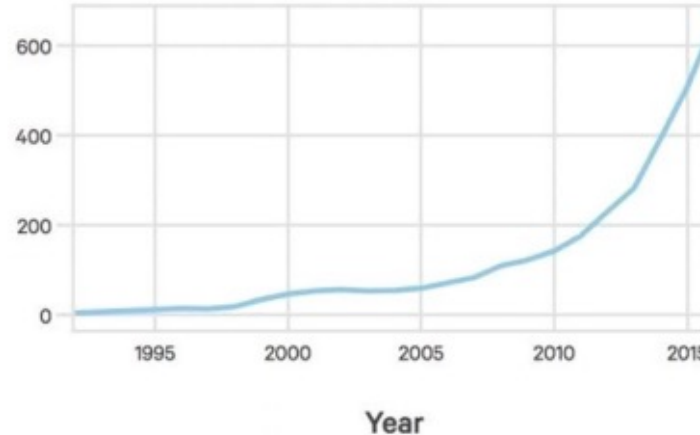
AI's Explosive Growth & Impact

Attendance at large conferences (1984-2019)
Source: Conference provided data.



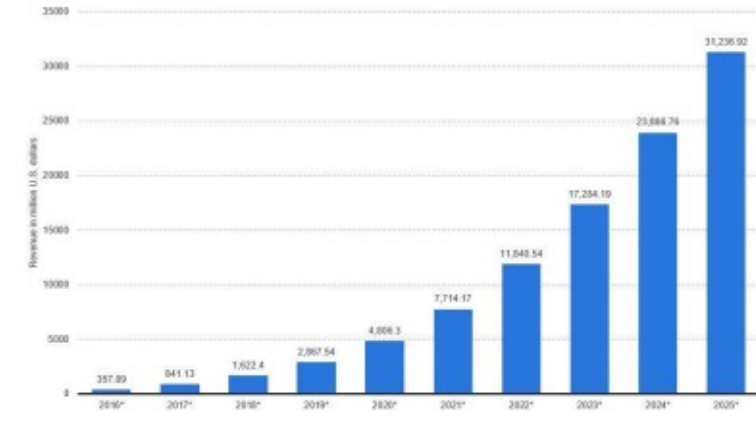
Number of attendance At AI conferences

Source: The Gradient



Startups Developing AI Systems

Source: Crunchbase, VentureSource, Sand Hill Econometrics



Enterprise Application AI Revenue

Source: Statista

Language: Transformer

- In 2017, transformer was introduced for NLP tasks.
- Achieved state-of-the-art results on eleven NLP tasks
- Pre-trained transformers (like BERT) can be fine-tuned for a wide range of tasks, such as question answering and language inference
 - without substantial task-specific architecture modifications
 - with just one additional output layer

Self-supervised Models

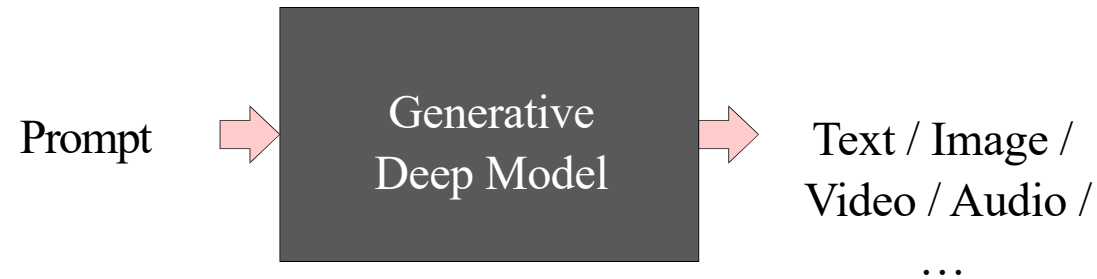
- How do we use unlabeled data for learning representations?
 - Predict next word / patch of image
 - Predict missing word / patch of image
 - Predict if two images are related (contrastive learning)
- Making powerful foundation models using this approach

Multi-modal Models

- Using the available large amount of multi-modal data
- CLIP: Learns a multi-modal embedding space by jointly training an image encoder and text encoder
- Zero-shot classification

Generative Models

Output text, image, video, audio, ... given no condition or a partial guidance or prompt

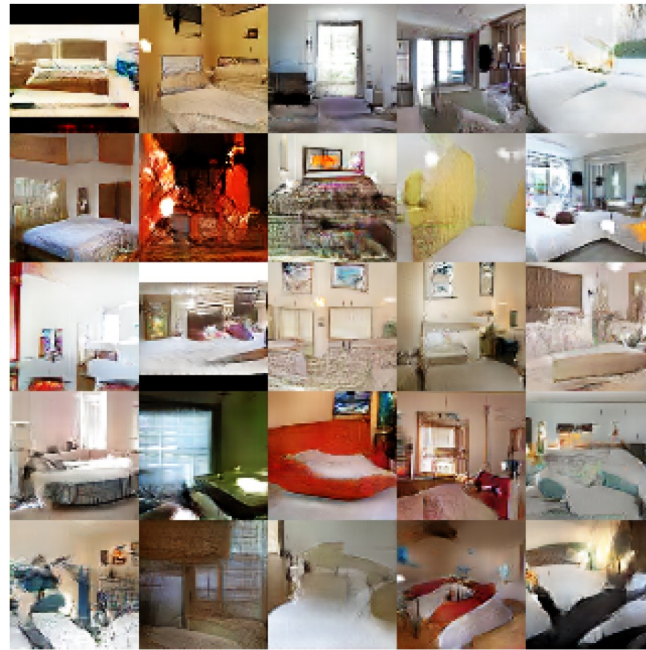


Generate samples by GAN



CIFAR-10

Goodfellow et al., GAN, NIPS 2014



LSUN

Radford et al., DCGAN, ICLR 2016



CelebA

Karras et al., Progressive GAN, ICLR 2018



ImageNet

Brocks et al., BigGAN, ICLR 2019

Denoising Diffusion Models

Emerging as powerful generative models, outperforming GANs



[Dhariwal & Nichol, Diffusion Models Beat GANs, OpenAI, 2021](#)



[Ho et al., Cascaded Diffusion Models, Google, 2021](#)

DALL-E2 & Imagen

OpenAI DALL·E 2

a teddy bear on a skateboard in times square



[Ramesh et al., Hierarchical Text-Conditional Image Generation with CLIP Latents, 2022](#)

Google Imagen

A group of teddy bears in suit in a corporate office celebrating the birthday of their friend. There is a pizza cake on the desk.



[Saharia et al., Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding, 2022](#)

OpenAI GPT3: few-shot and zero-shot learning

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 cheese => ..... ← prompt
```

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 sea otter => loutre de mer ← example
3 cheese => ..... ← prompt
```

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
1 Translate English to French: ← task description
2 sea otter => loutre de mer ← examples
3 peppermint => menthe poivrée ←
4 plush girafe => girafe peluche ←
5 cheese => ..... ← prompt
```

Large Language Models (LLMs)

- Large language models are one of the most successful applications of transformer models.
- Using massive unlabeled text datasets to learn LLMs (as self-supervised pre-trained models)
 - recognize, summarize, translate, predict and generate text and other content based on knowledge gained from massive datasets.

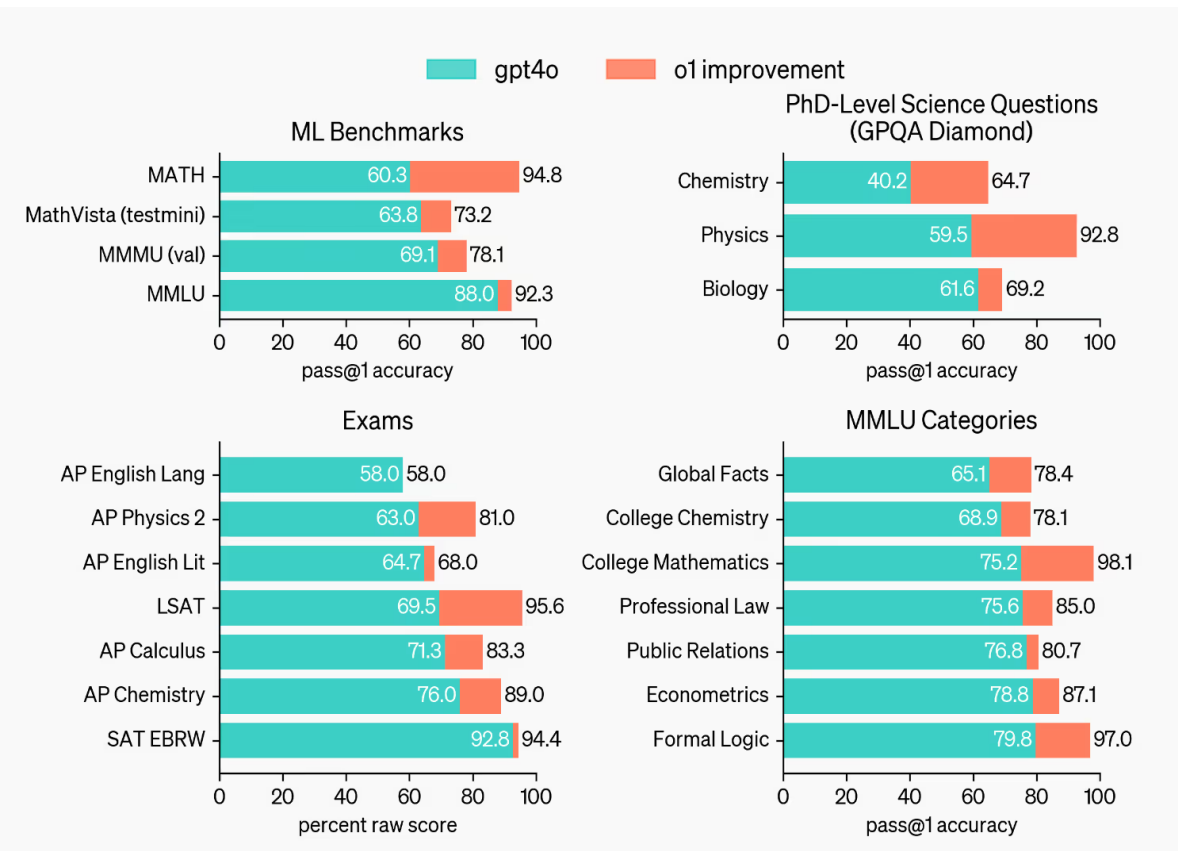
Recent Large Language Models (LLMs)

Conversational generative AI chatbots:

- generate response to a wide range of prompts and questions
- answer questions, translate languages, write different kinds of creative content, generate codes, provide summaries of topics or create stories

Tracking Progress

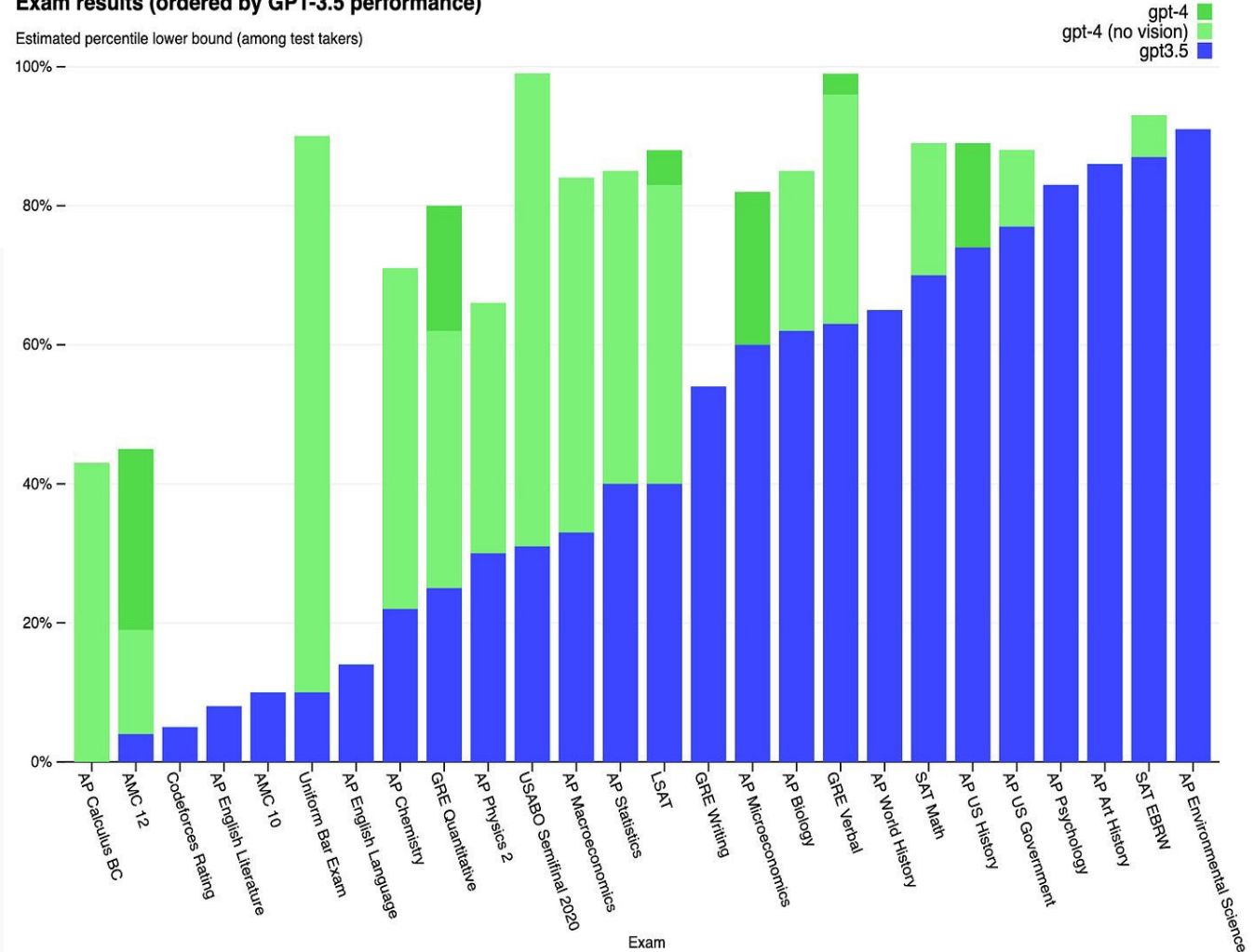
- How well AI can do human tasks



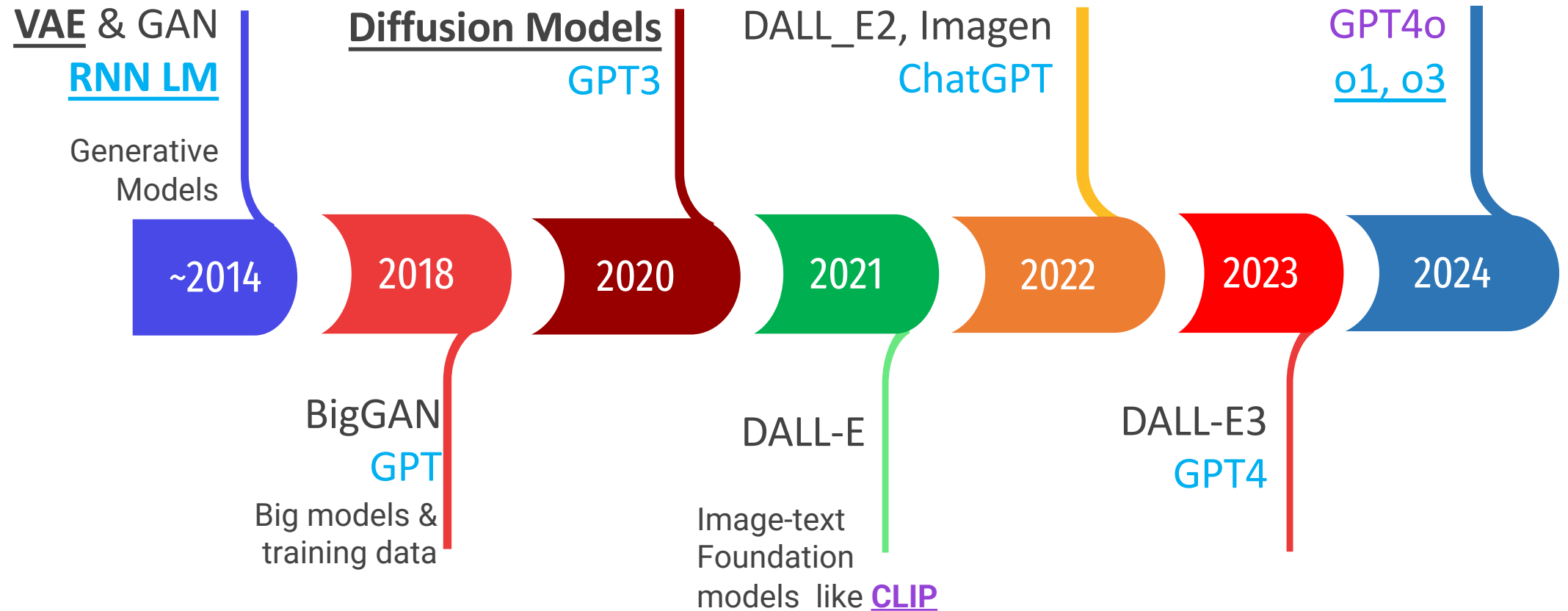
[OpenAI]

Exam results (ordered by GPT-3.5 performance)

Estimated percentile lower bound (among test takers)



Timeline of Generative Image/text models



40719 Course Objectives

- Understanding neural networks and training issues
- Comprehending several popular and also recent networks for various tasks
- Fearlessly design, develop, and train networks
 - Hands-on practical experience.
- Gain an understanding of where the field is

40719 Deep Learning

- Deep Learning Basics (Lectures 1-6)
- Deep Learning Architectures (Lectures 7-12, 26)
- Generative Models (Lectures 17-21)
- Foundation Models (Lectures 13-16, 22-25)

40719 Deep Learning

- Deep Learning Basics (Lectures 1-6)
 - MLP, back propagation, optimization and Generalization techniques, training issues and techniques
- Deep Learning Architectures (Lectures 7-12, 26)
 - CNN, RNN, Attention, Transformer, GNNs
- Generative Models (Lectures 17-21)
 - VAE, GAN, diffusion models
- Foundation Models (Lectures 13-16, 22-25)
 - LLMs
 - Visual models
 - Multi-modal models