introduction to artificial intelligence

brettkoonce.com/talks

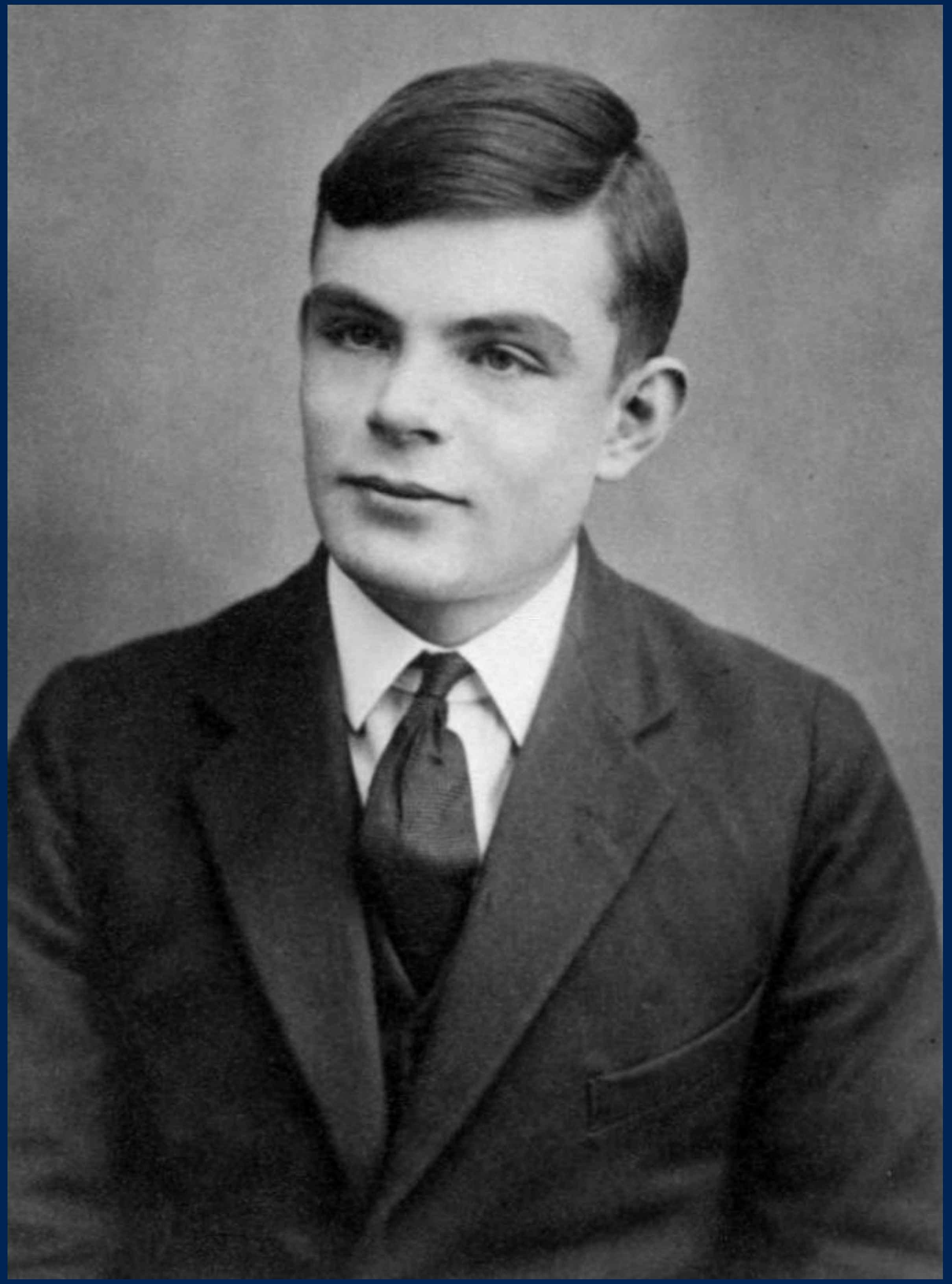
may 7, 2021

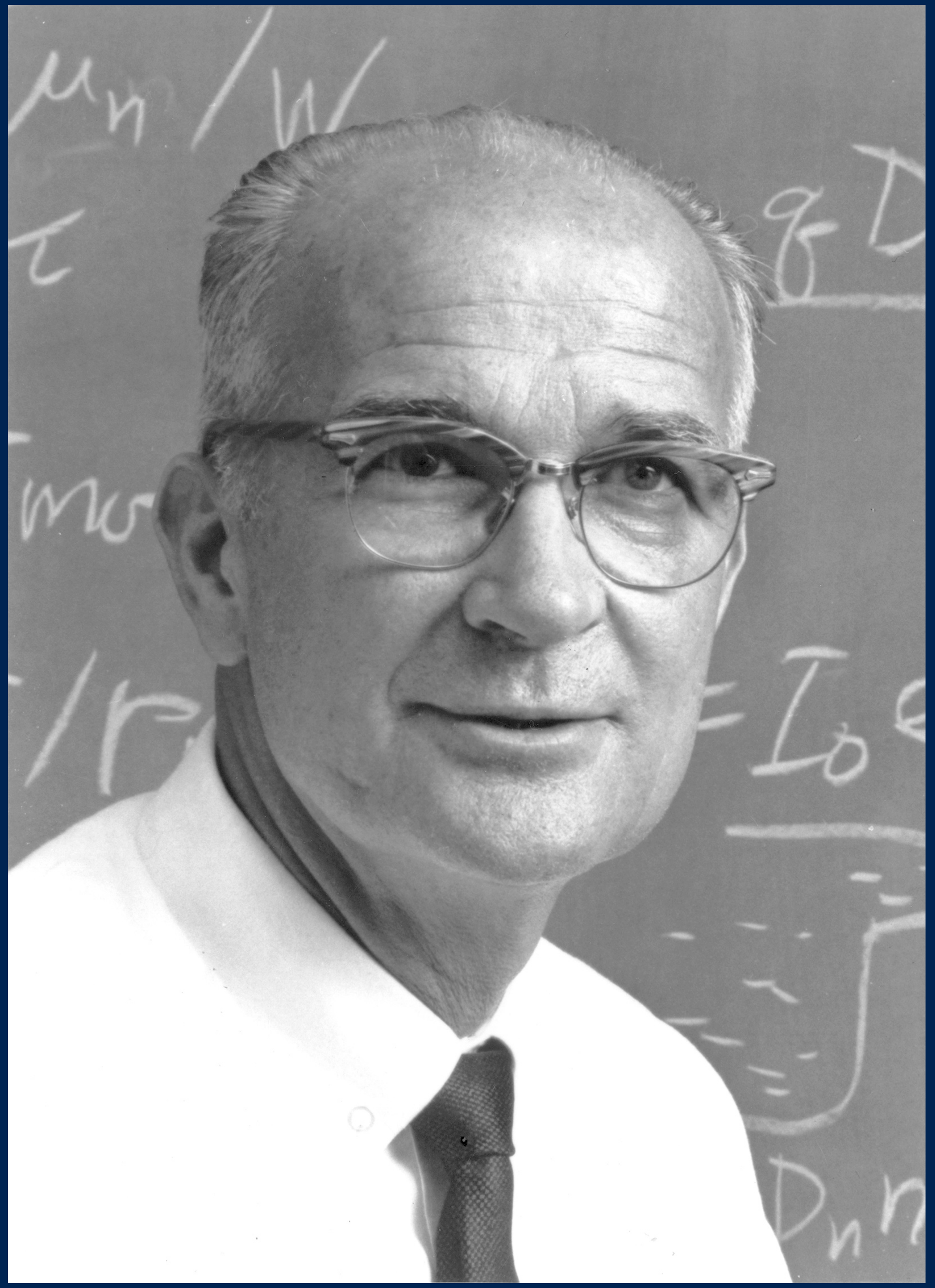
outline

- historical context
- neural networks
- applications
- scaling hypothesis
- future

prehistory

- turing machines
- transistors
- computers



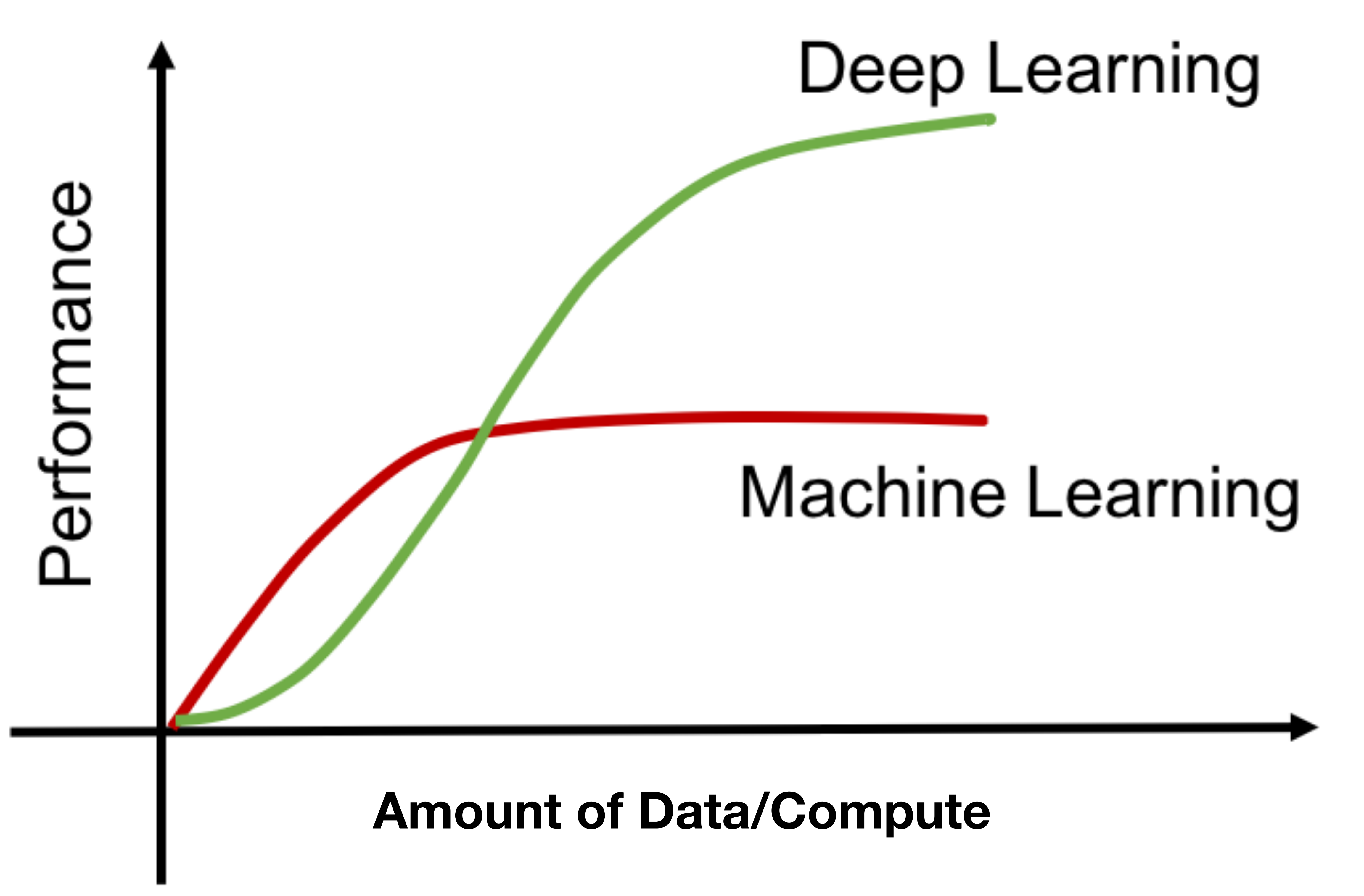


early ai

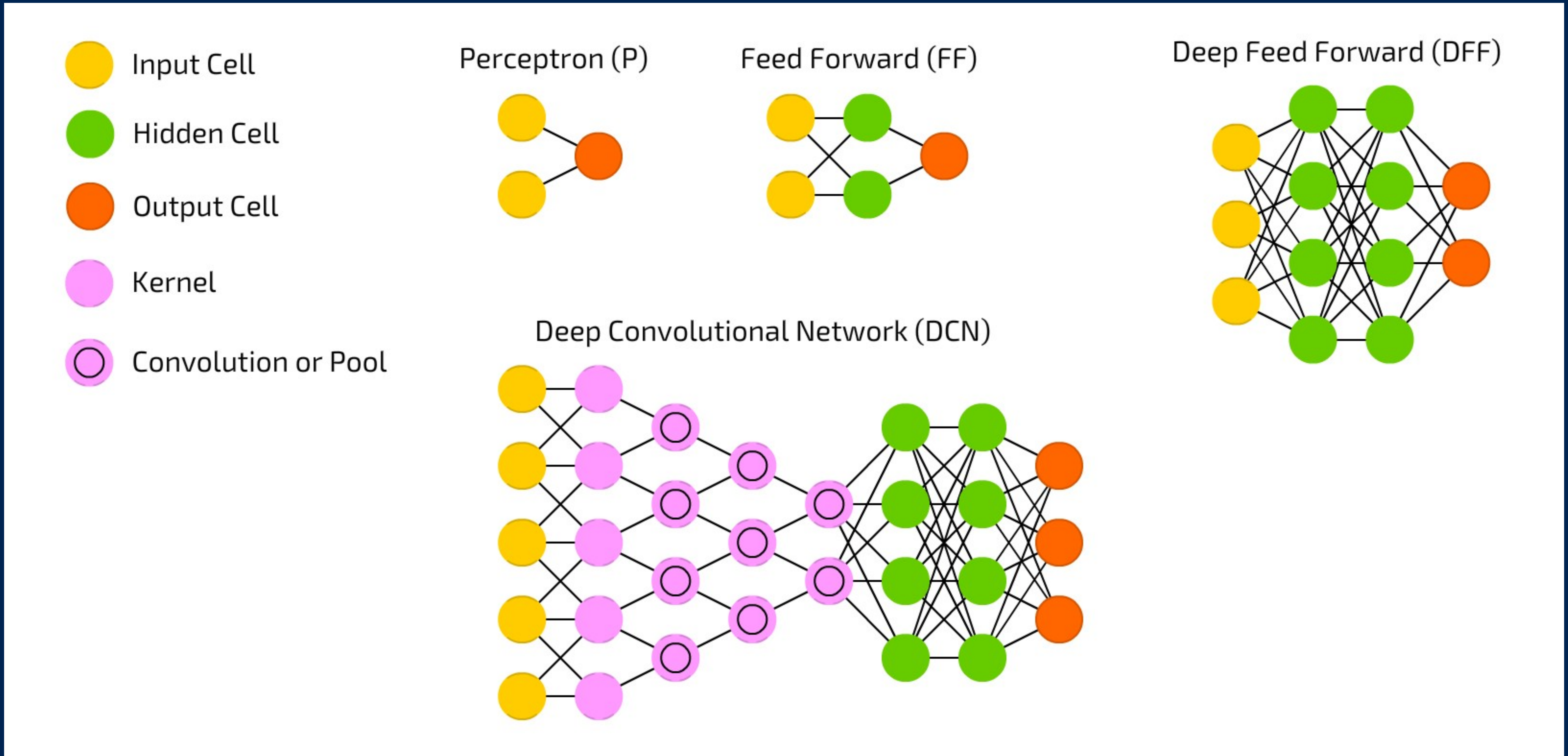
- symbolic logic
- statistical methods
- machine learning
- perceptron: 1959

machine <2010)

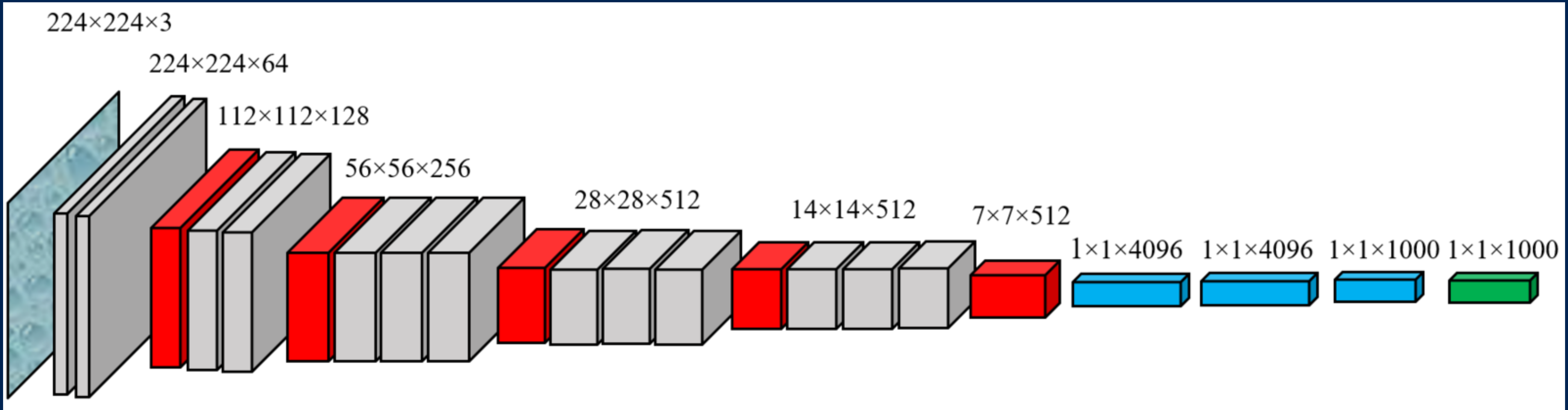
- large scale data
- large scale compute
- small scale algorithms



convolutional neural networks



vgg (2014)


224×224×3
224×224×64
112×112×128
56×56×256
28×28×512
14×14×512
7×7×512
1×1×4096
1×1×4096
1×1×1000
1×1×1000

resnet (2015)

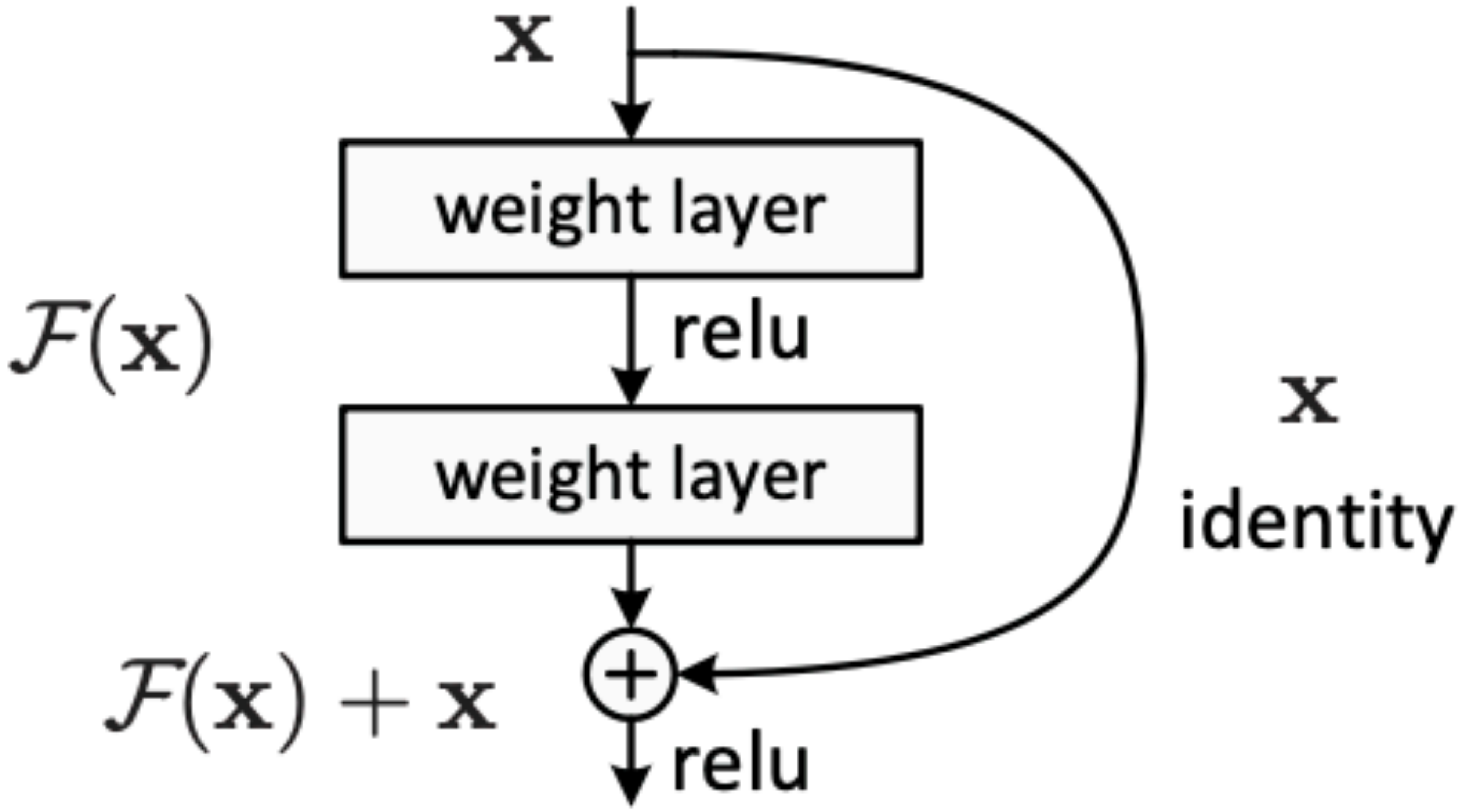
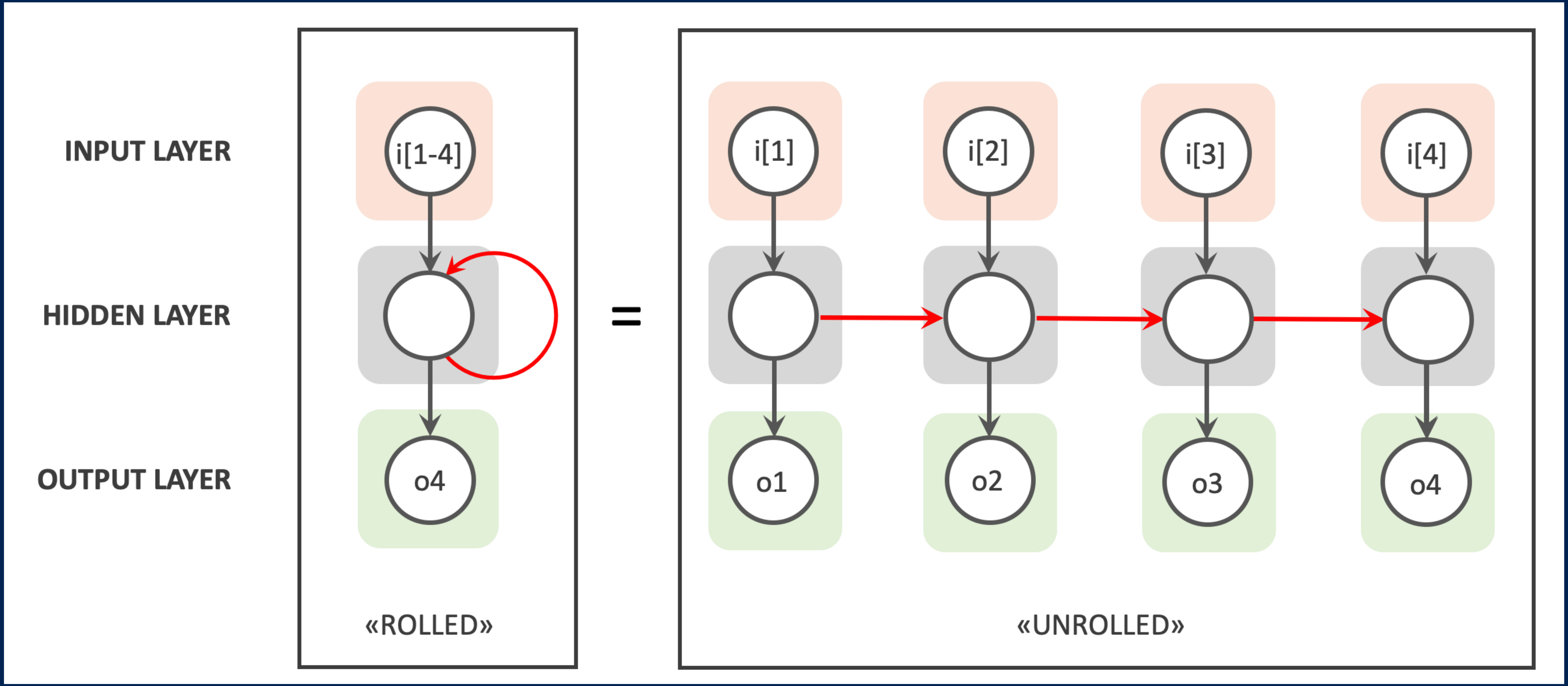
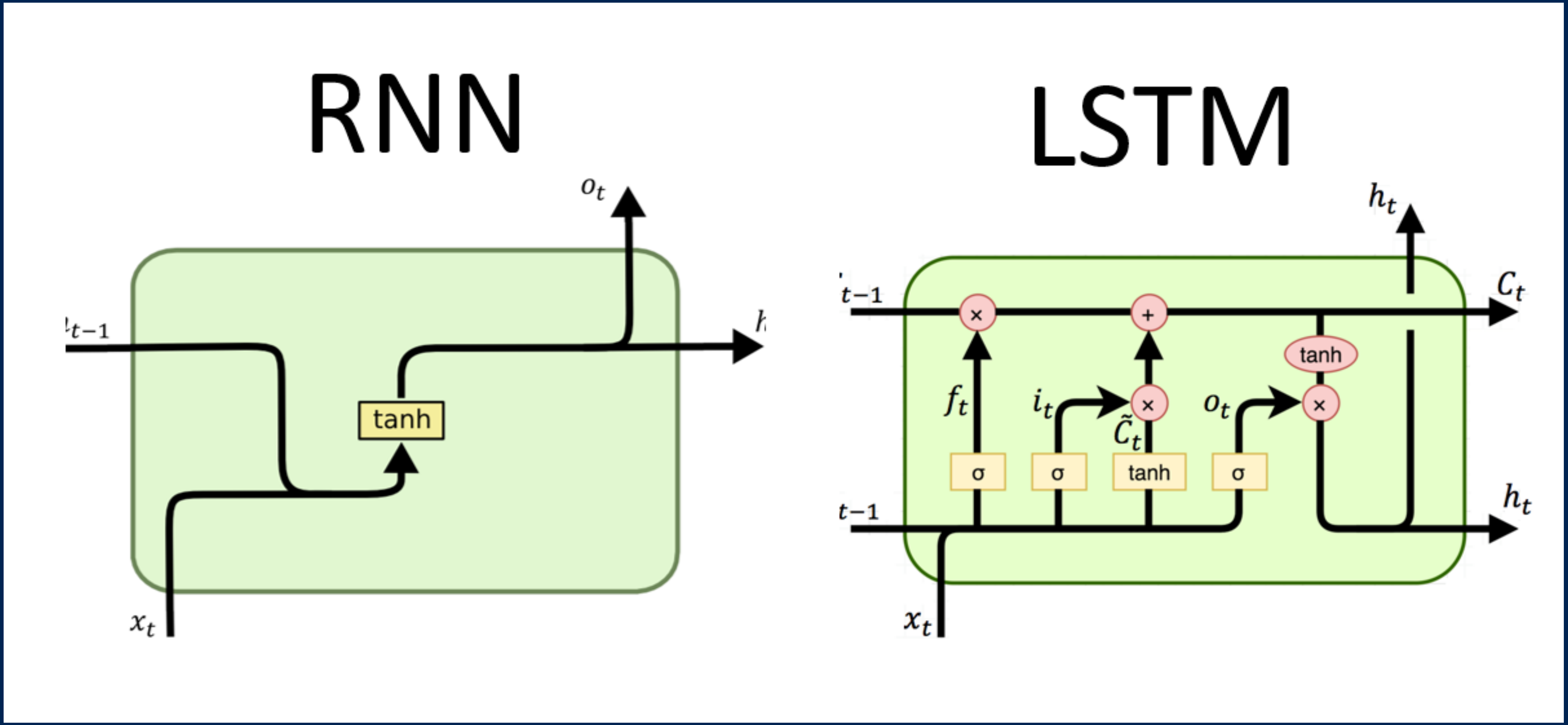


Figure 2. Residual learning: a building block.

recurrent neural networks (1986)



long-short term memory (1997)



seq2seq (2014)

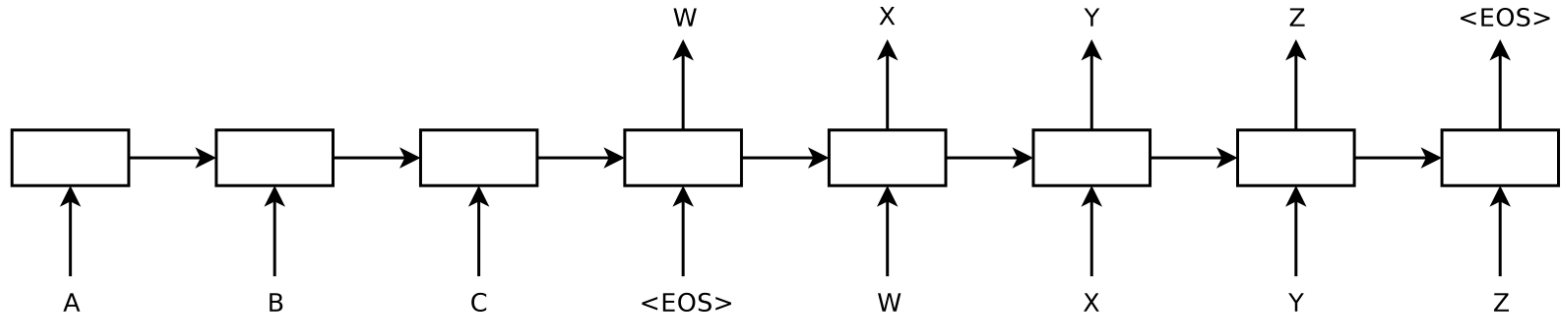


Figure 1: Our model reads an input sentence "ABC" and produces "WXYZ" as the output sentence. The model stops making predictions after outputting the end-of-sentence token. Note that the LSTM reads the input sentence in reverse, because doing so introduces many short term dependencies in the data that make the optimization problem much easier.

transformers (2017)

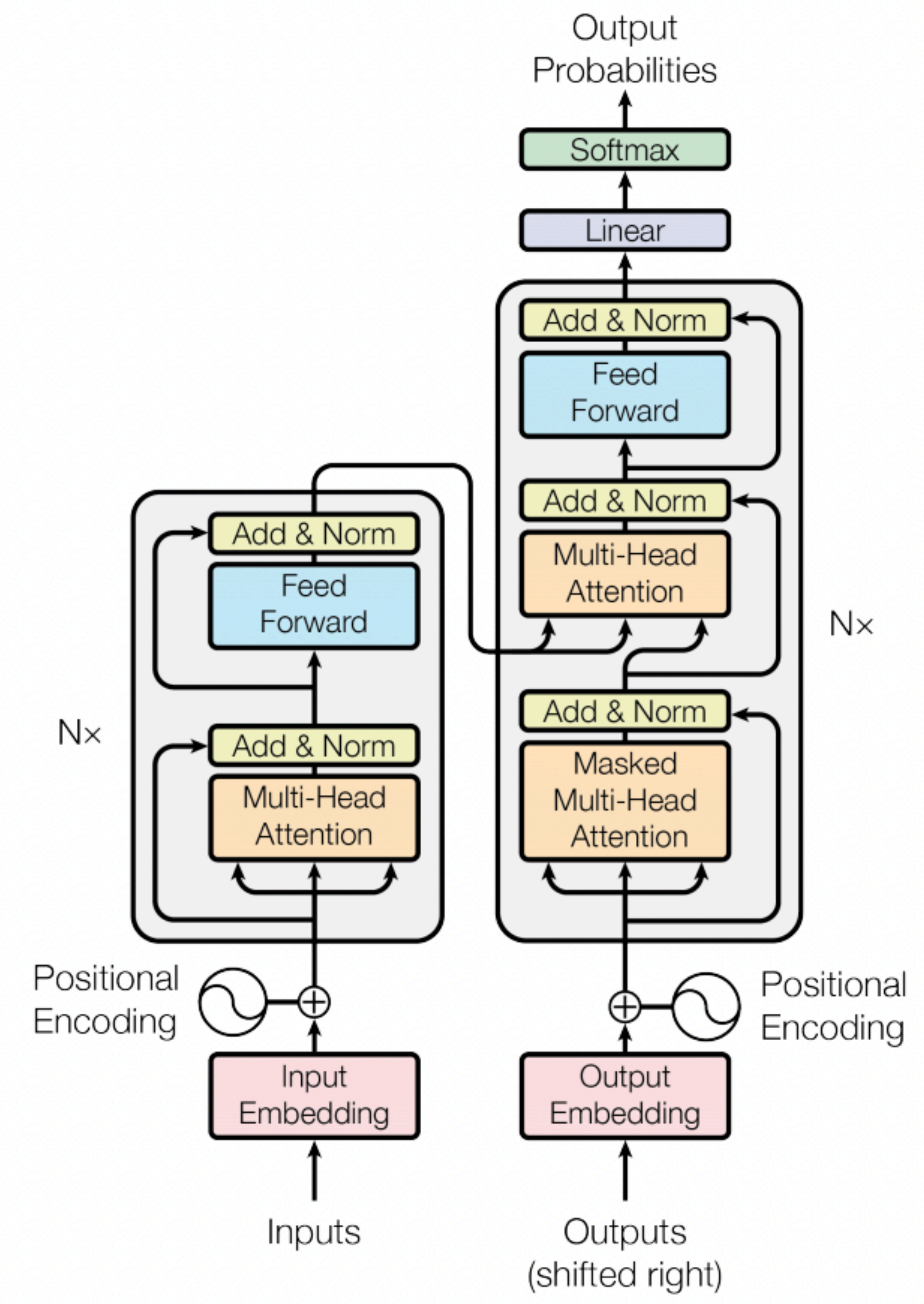
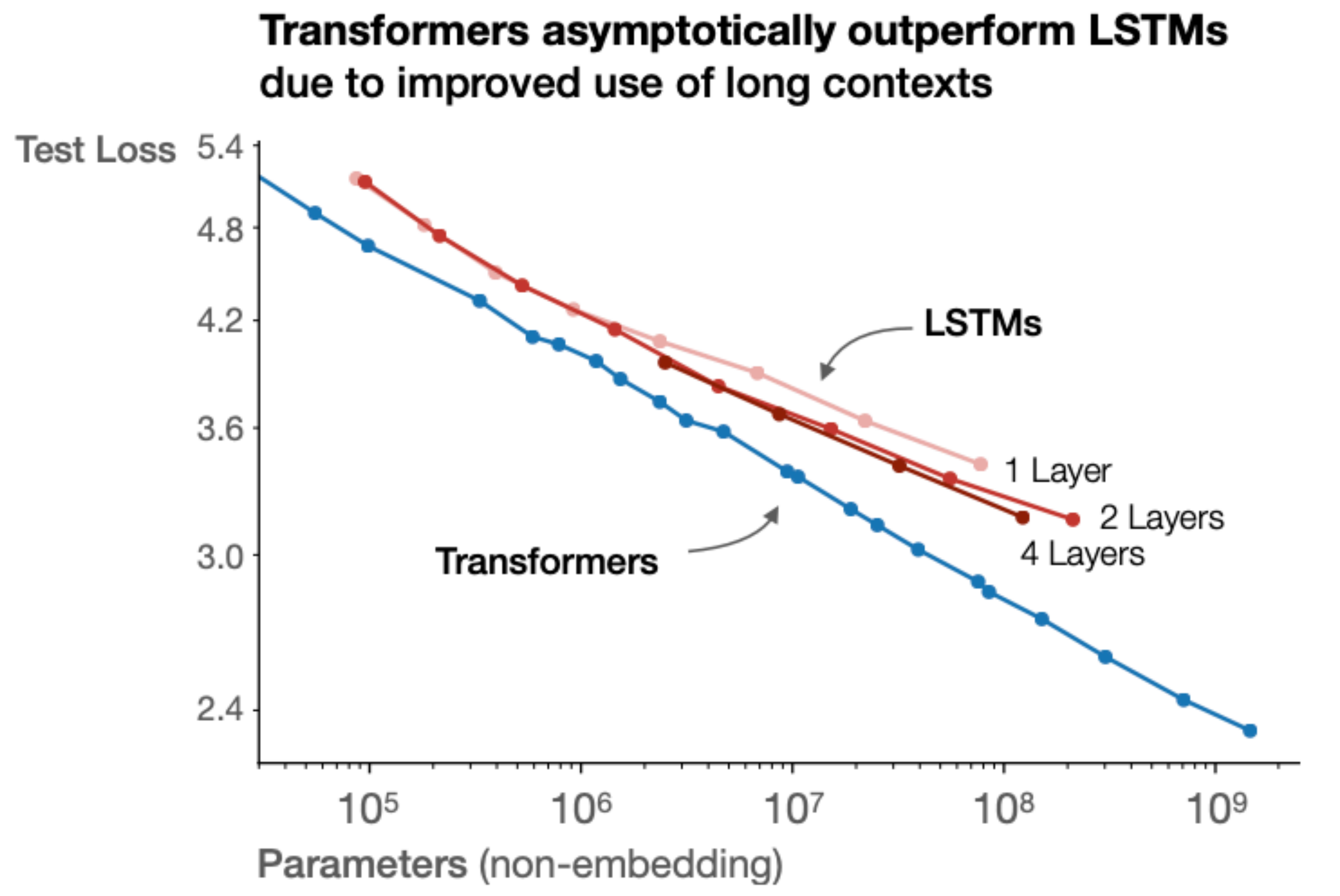


Figure 1: The Transformer - model architecture.



hybrid: mdetr (2021)

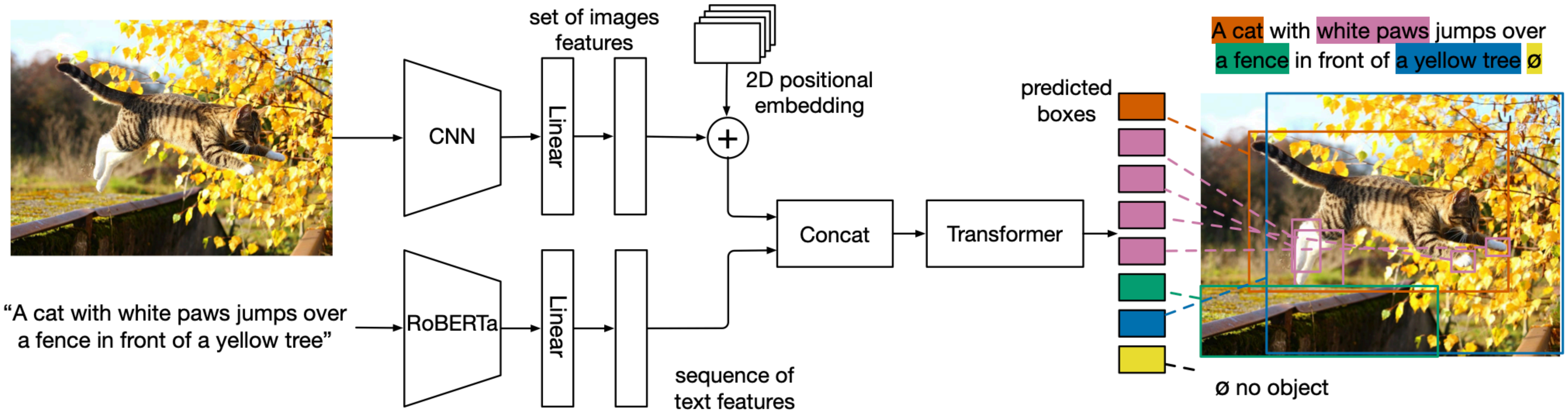
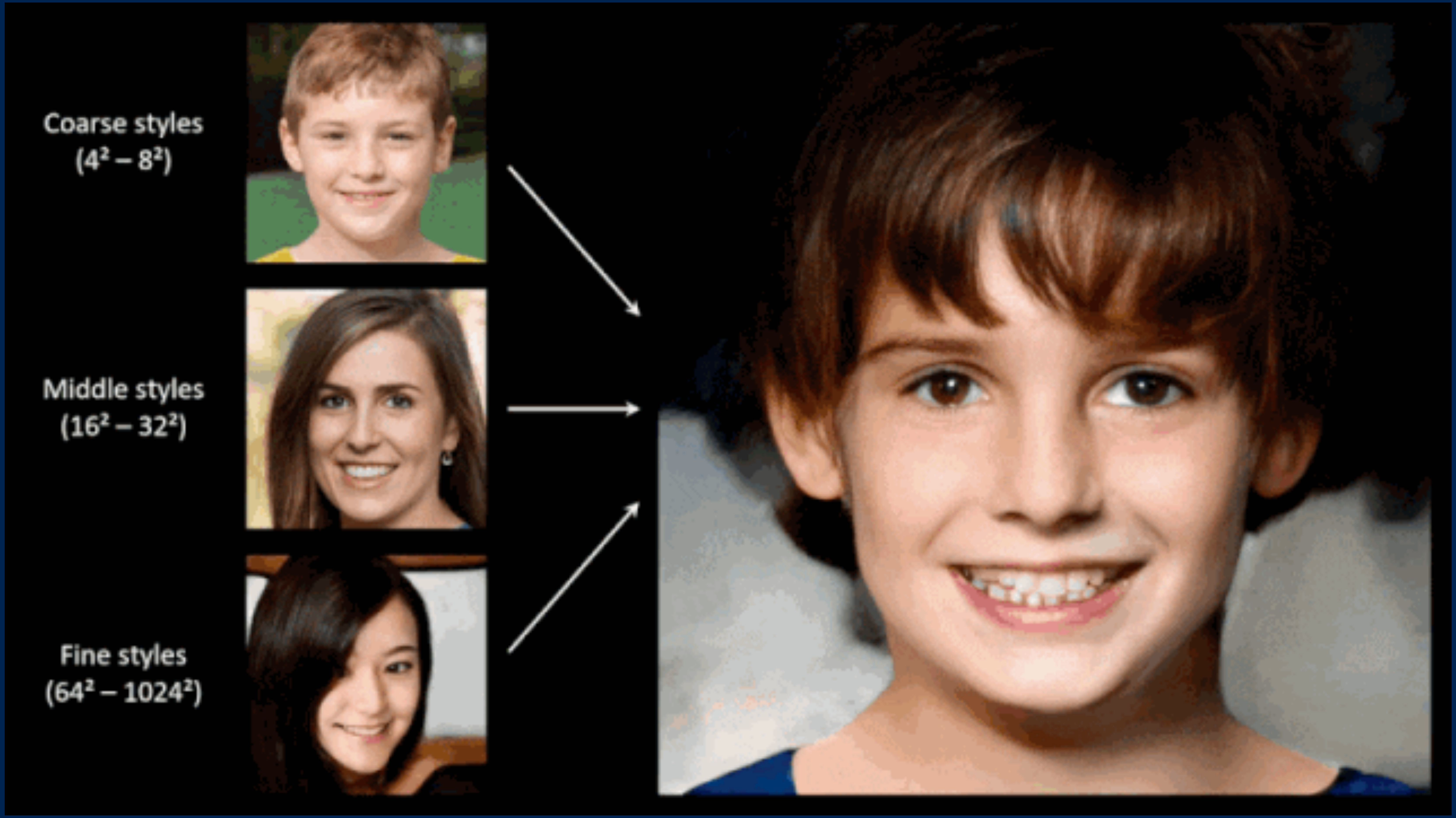
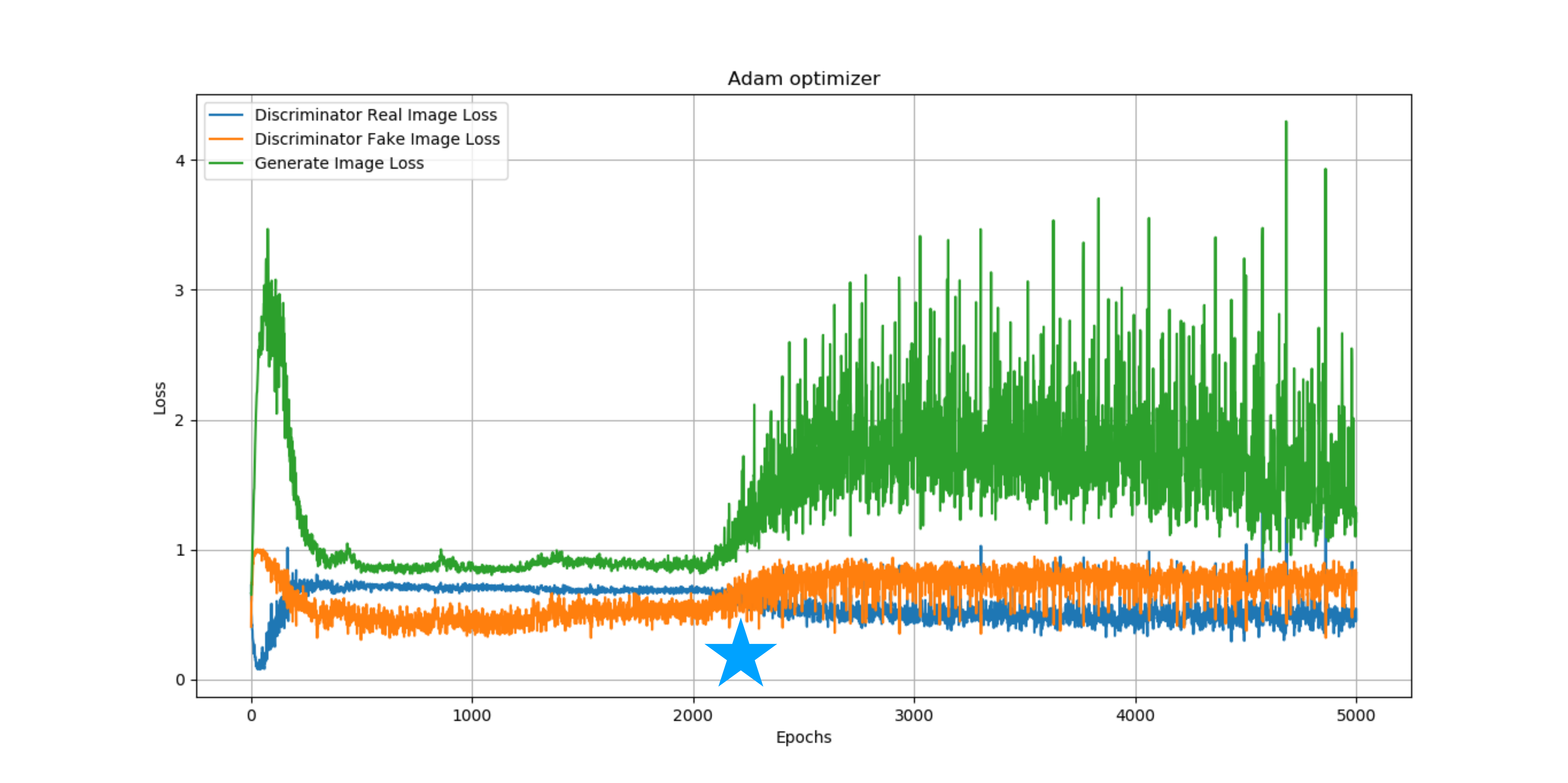


Figure 2: MDETR uses a convolutional backbone to extract visual features, and a language model such as RoBERTa to extract text features. The features of both modalities are projected to a shared embedding space, concatenated and fed to a transformer encoder-decoder that predicts the bounding boxes of the objects and their grounding in text.

generative adversarial networks (2014, stylegan: 2018)



gan training loss



reinforcement learning (dqn, 2015)

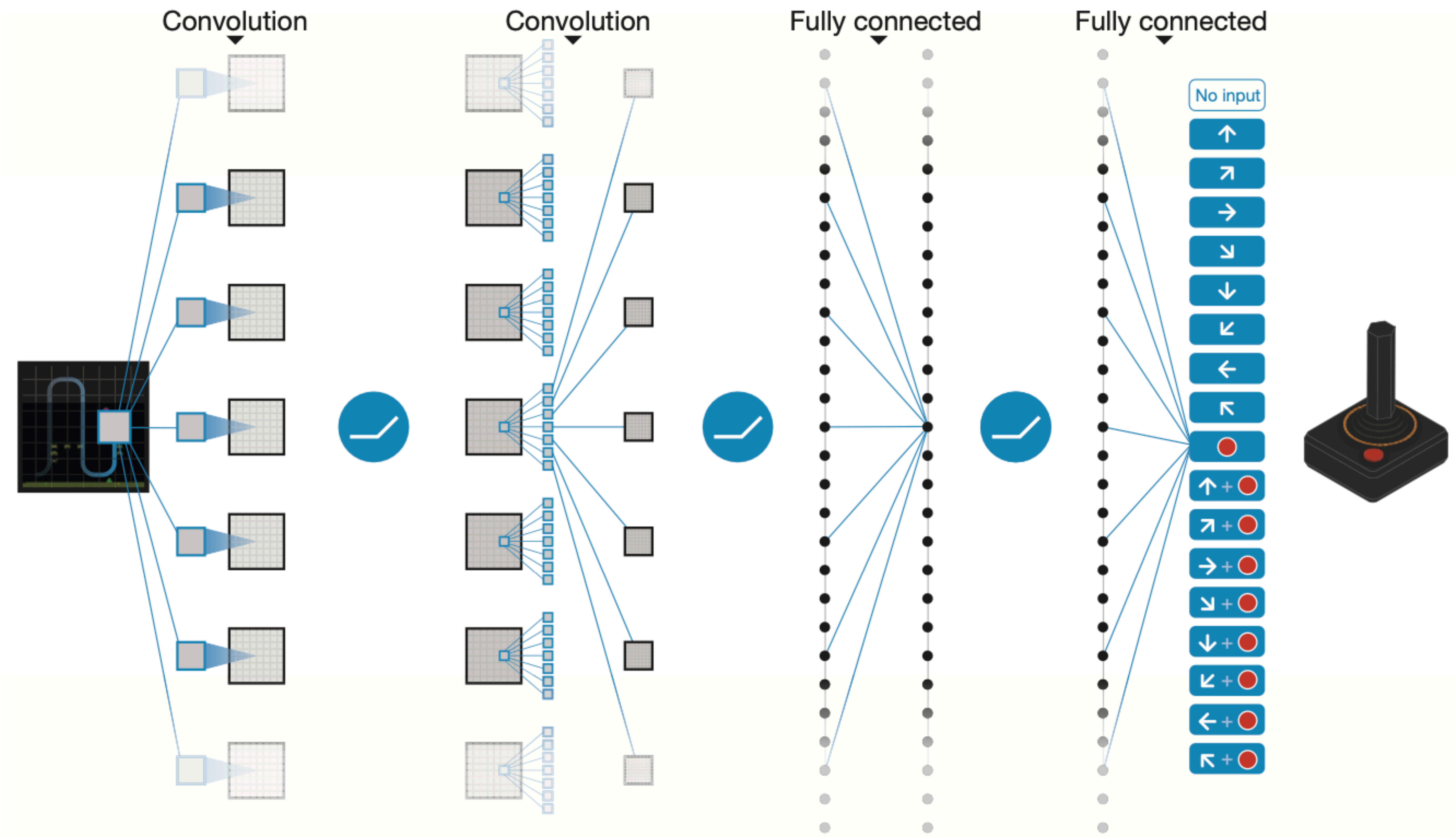
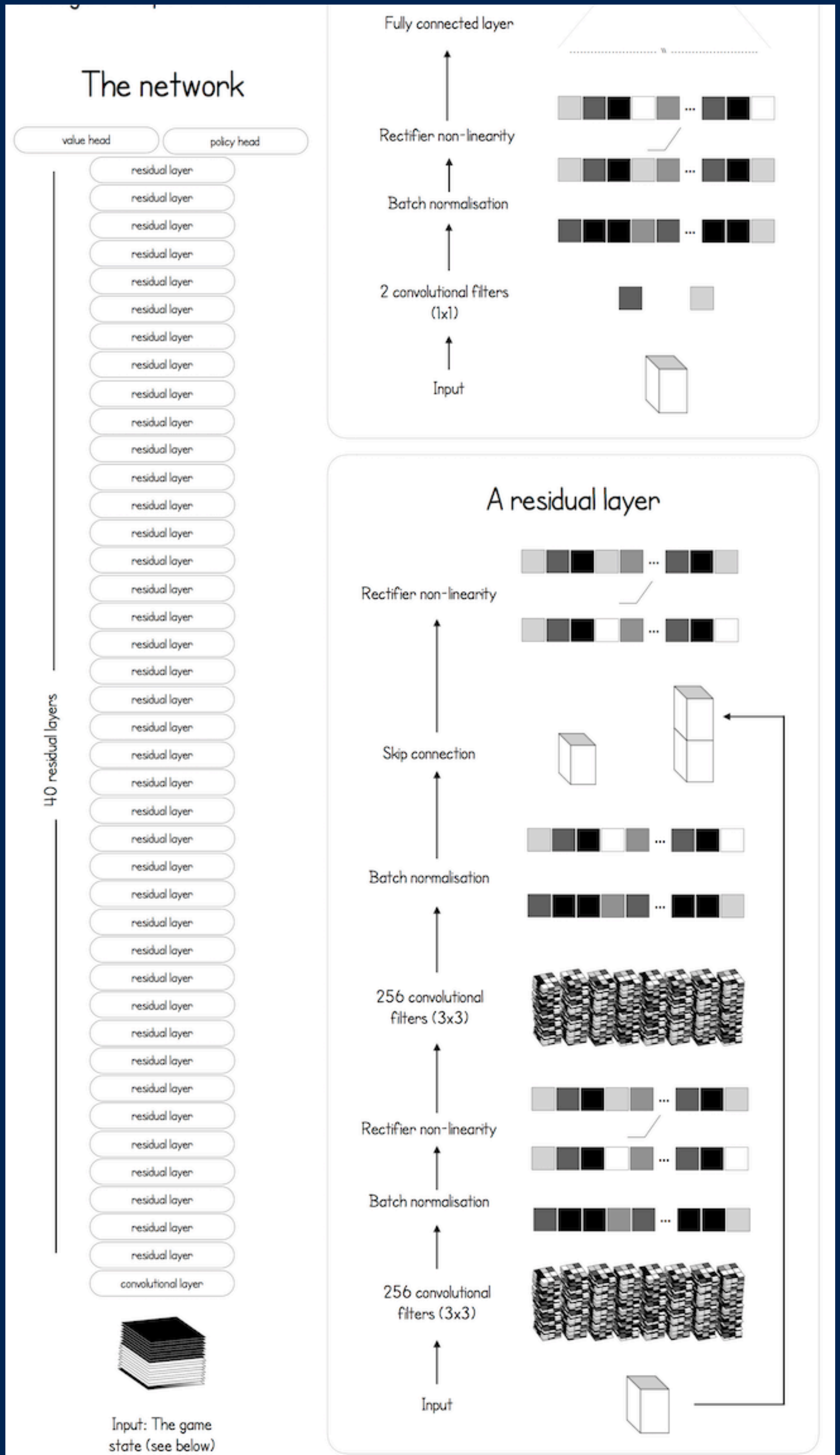
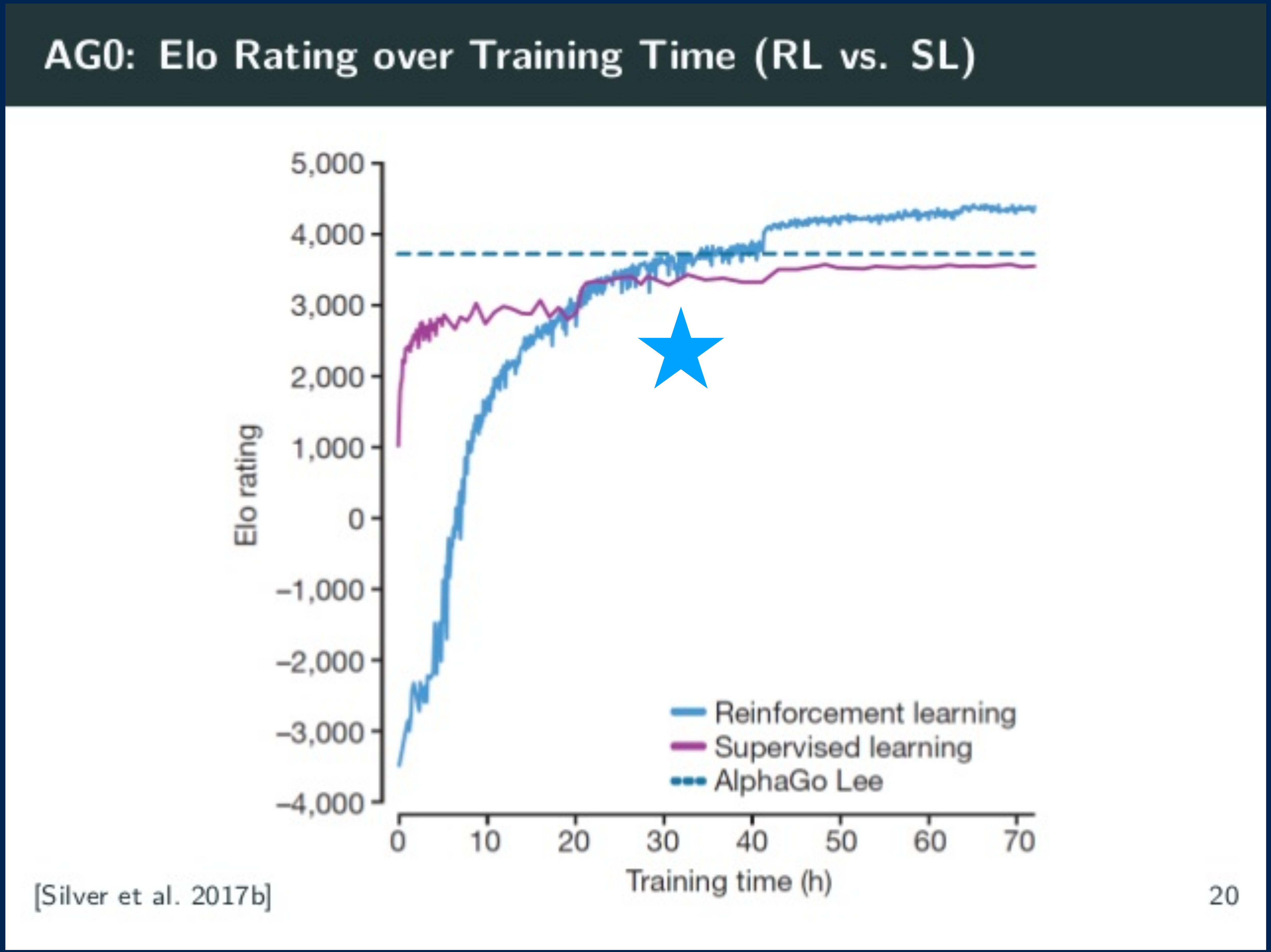


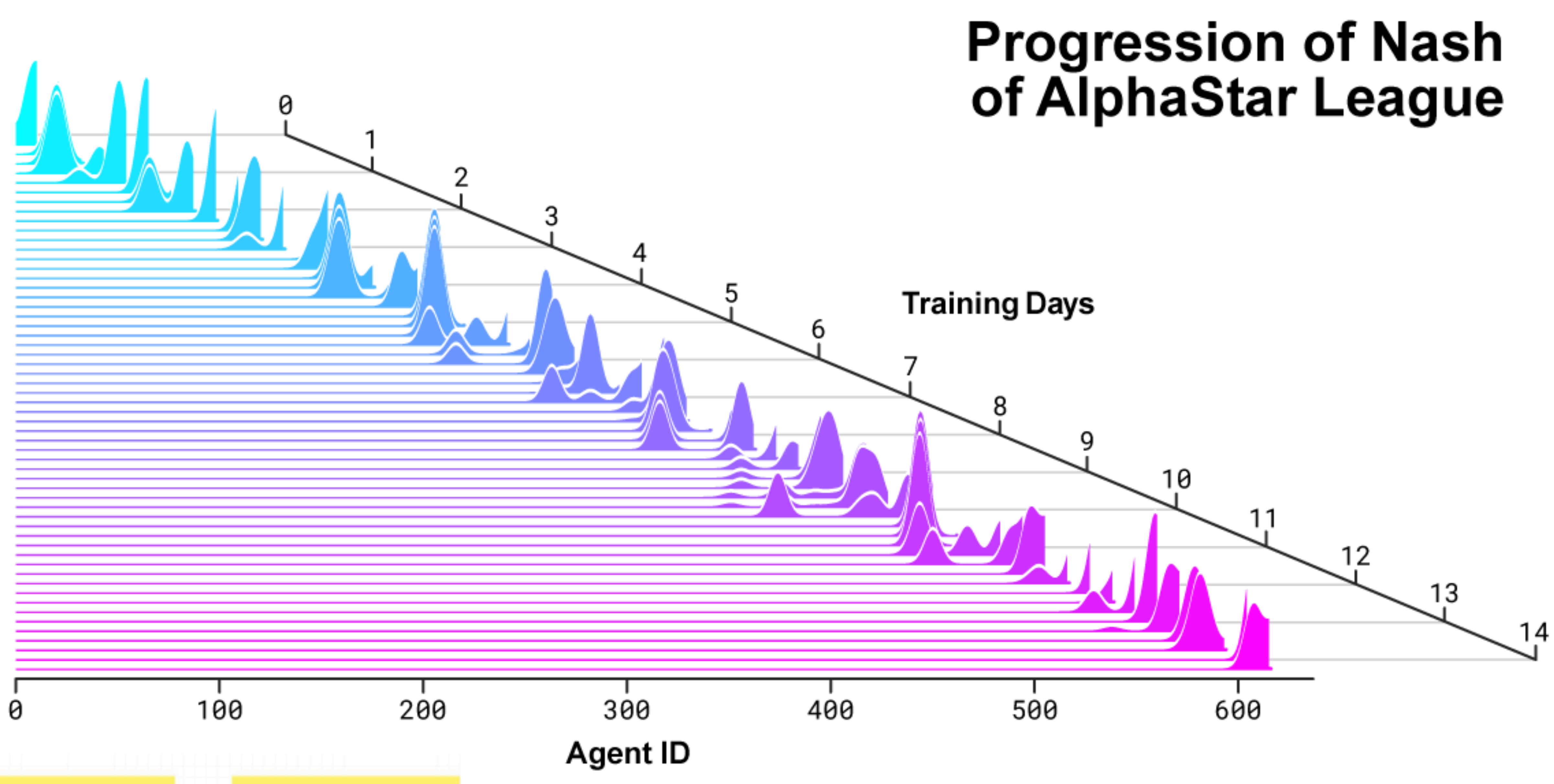
Figure 1 | Schematic illustration of the convolutional neural network. The details of the architecture are explained in the Methods. The input to the neural network consists of an $84 \times 84 \times 4$ image produced by the preprocessing map ϕ, followed by three convolutional layers (note: snaking blue line symbolizes sliding of each filter across input image) and two fully connected layers with a single output for each valid action. Each hidden layer is followed by a rectifier nonlinearity (that is, $\max(0,x)$).

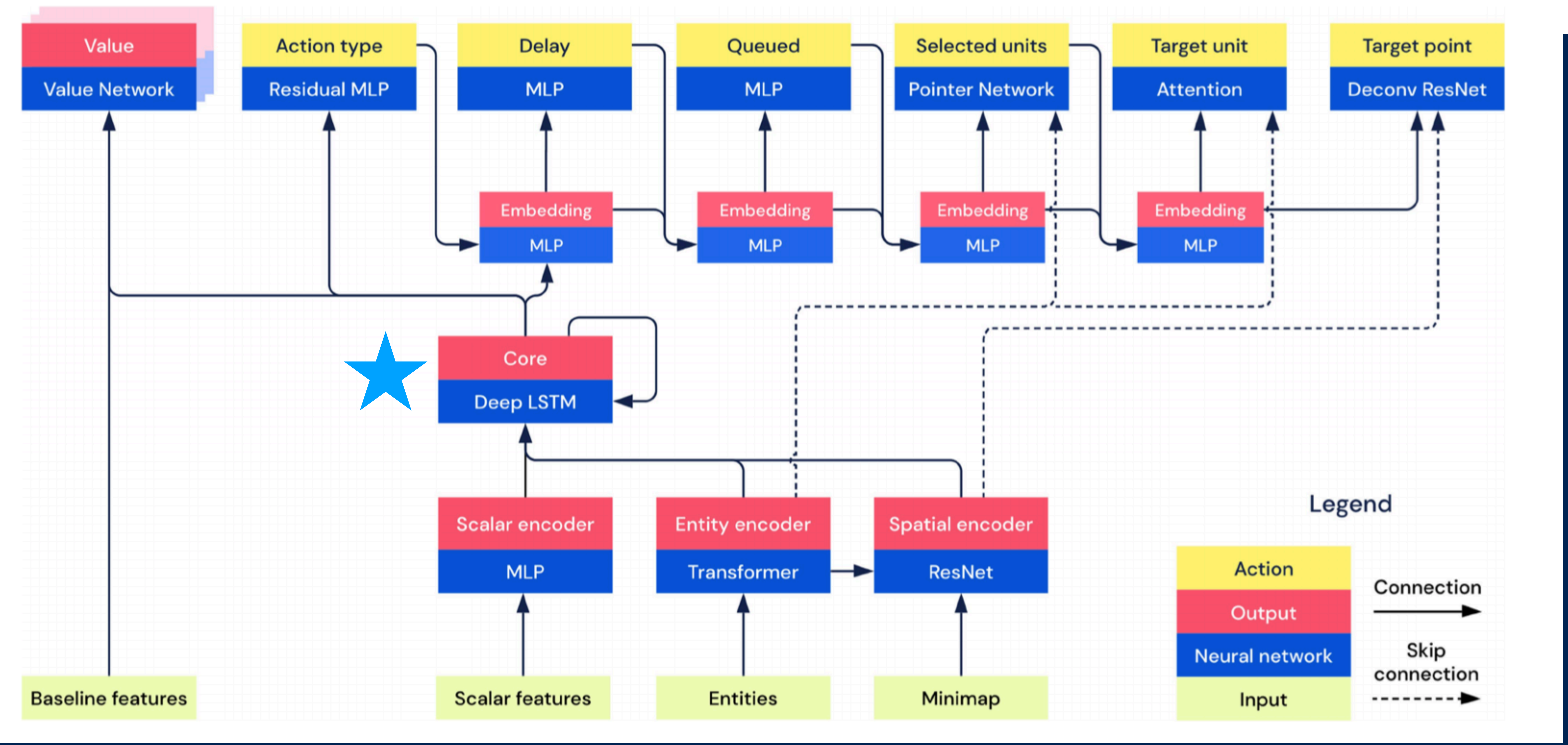
alphazero (2018)



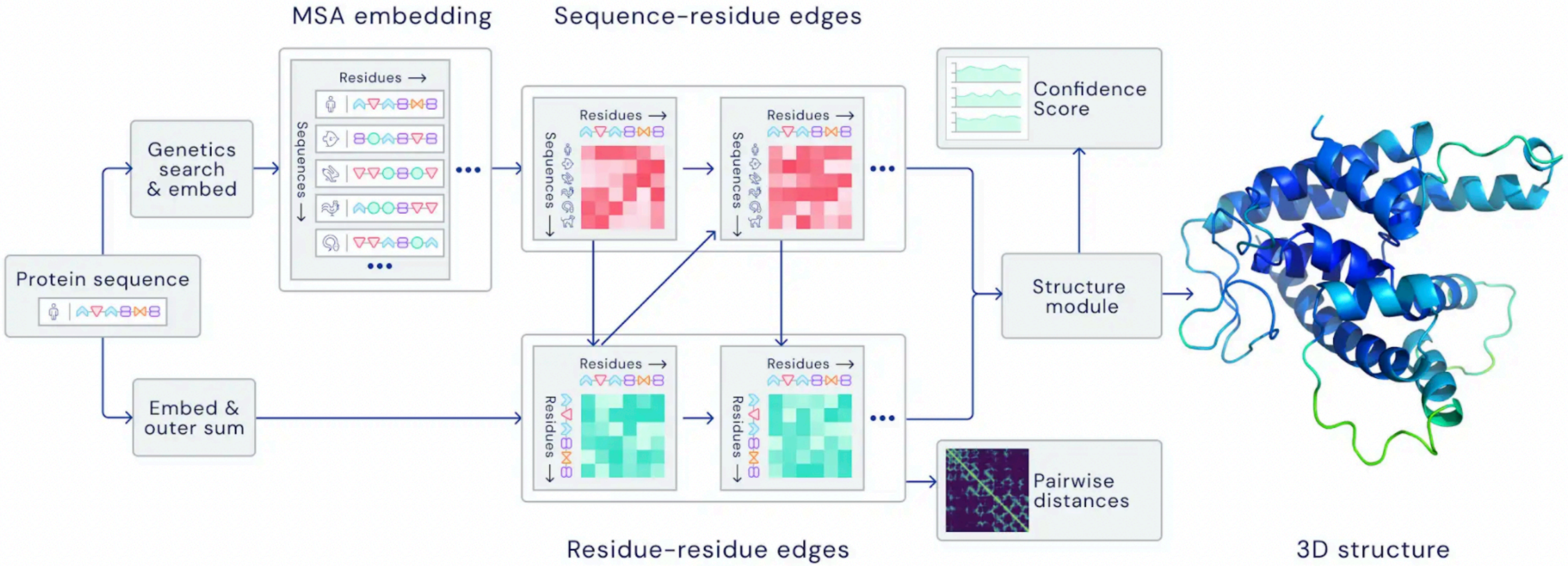


alphastar (2019)



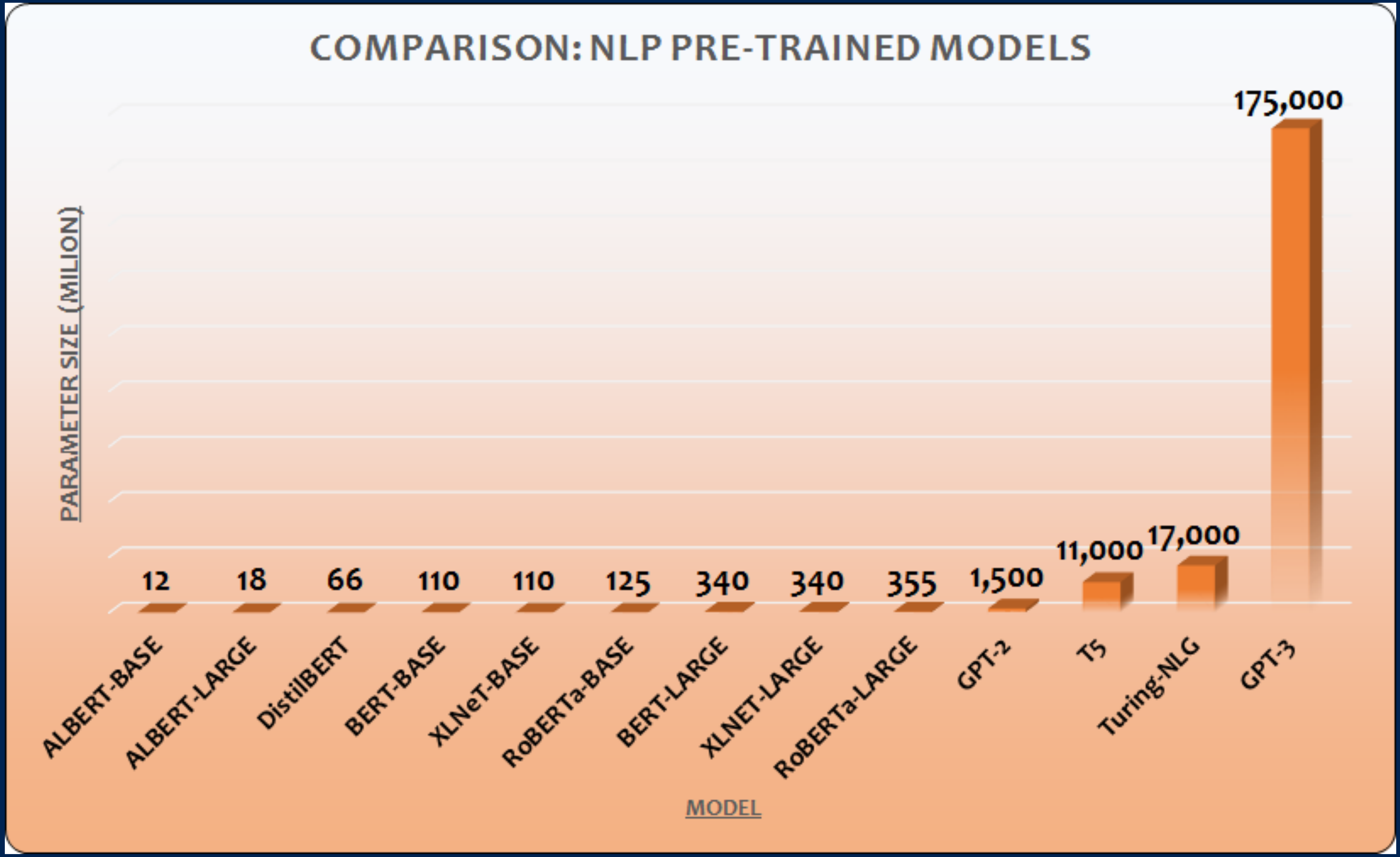


alphafold2 (2020)



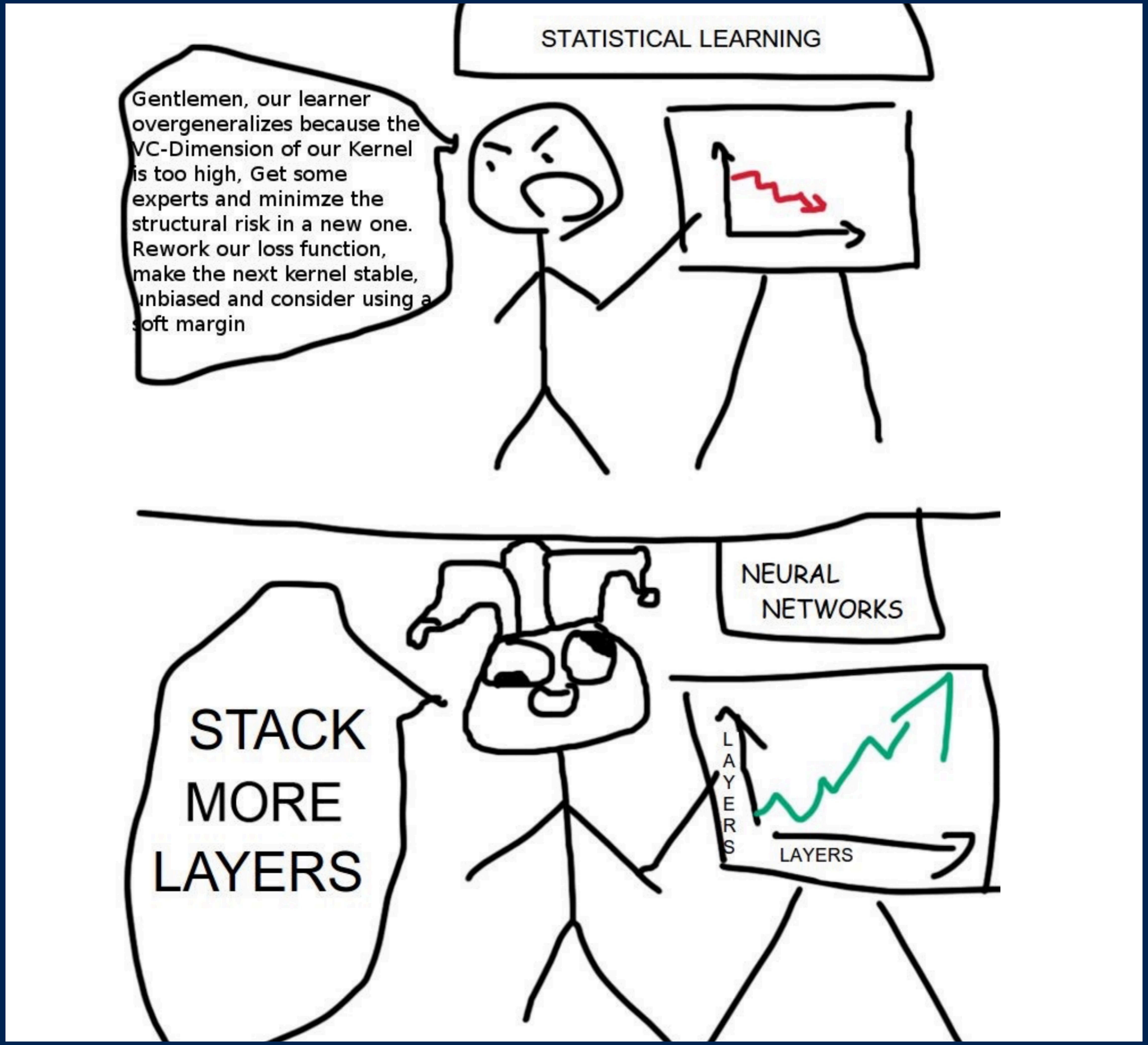
An overview of the main neural network model architecture. The model operates over evolutionarily related protein sequences as well as amino acid residue pairs, iteratively passing information between both representations to generate a structure.

gpt-3 (2020)



COMPARISON: NLP PRE-TRAINED MODELS
PARAMETER SIZE (MILION)
12
18
66
110
110
125
340
340
355
1,500
11,000
17,000
175,000
ALBERT-BASE
ALBERT-LARGE
DistilBERT
BERT-BASE
XLNeT-BASE
RoBERTa-BASE
BERT-LARGE
XLNET-LARGE
RoBERTa-LARGE
GPT-2
T5
Turing-NLG
GPT-3
MODEL

bitter lesson (sutton)



scaling hypothesis

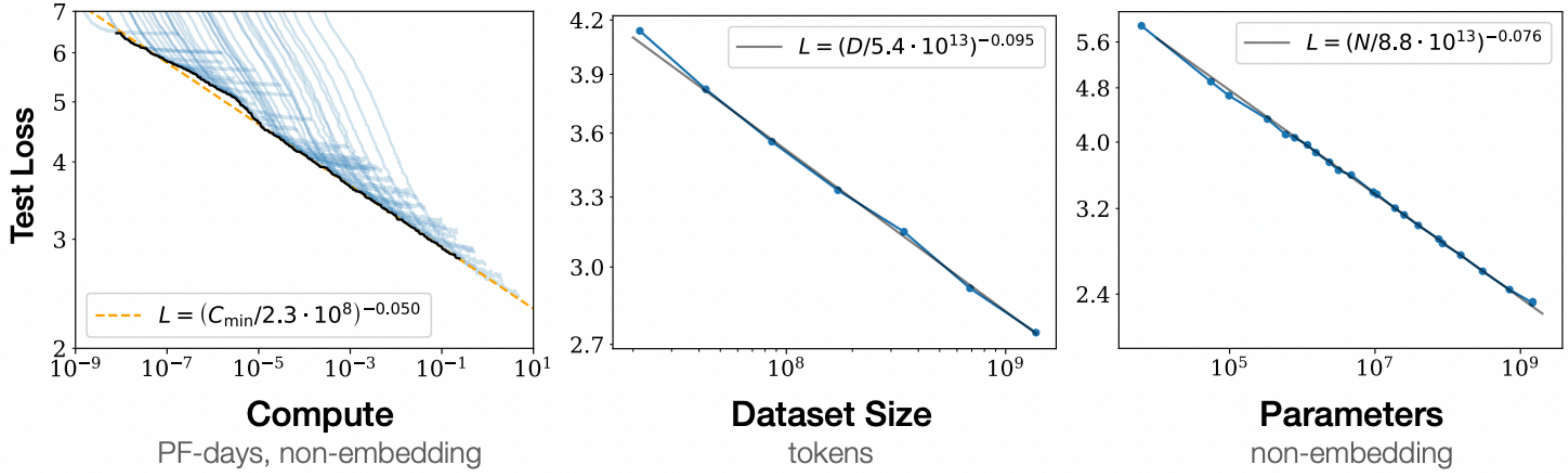
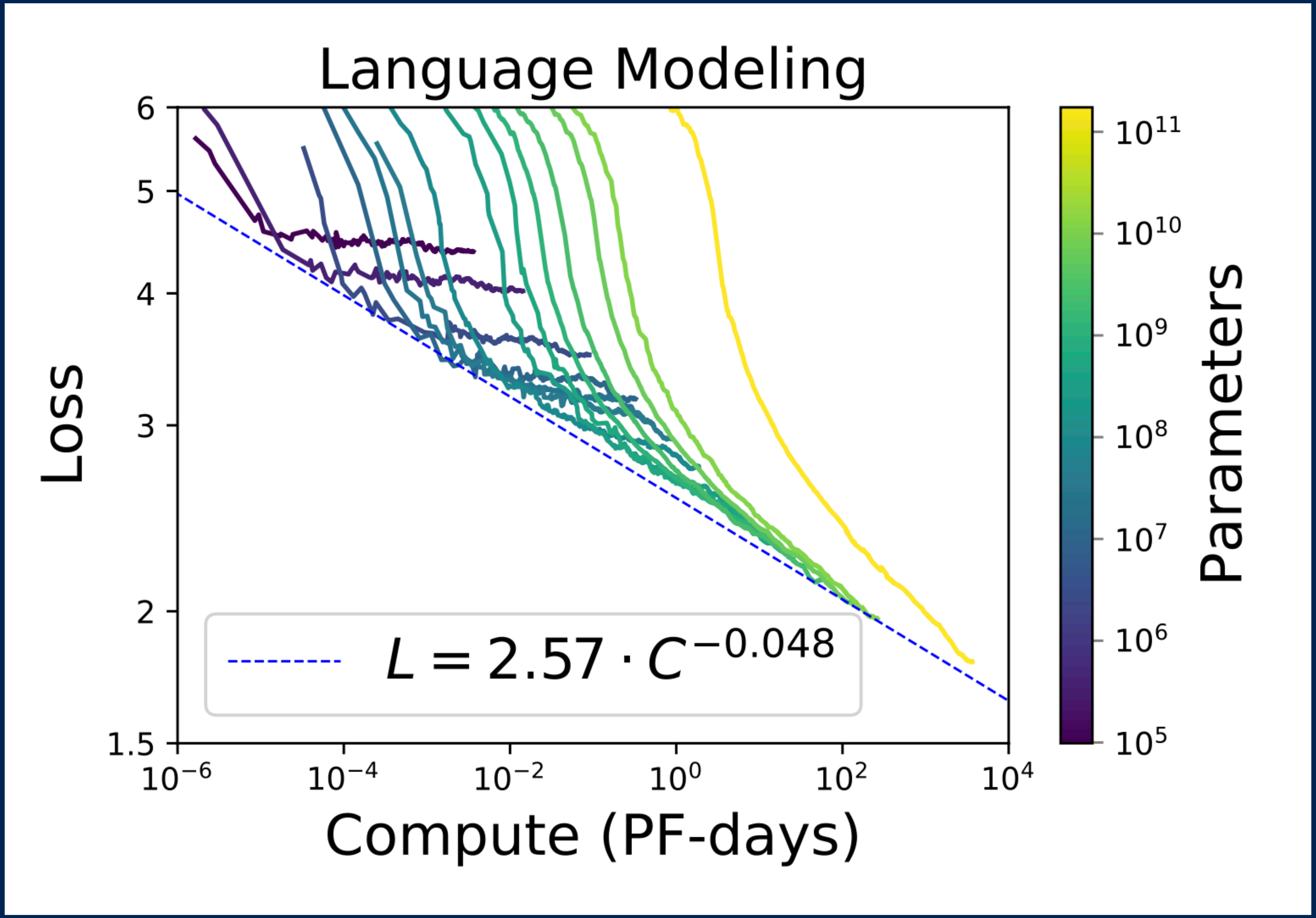
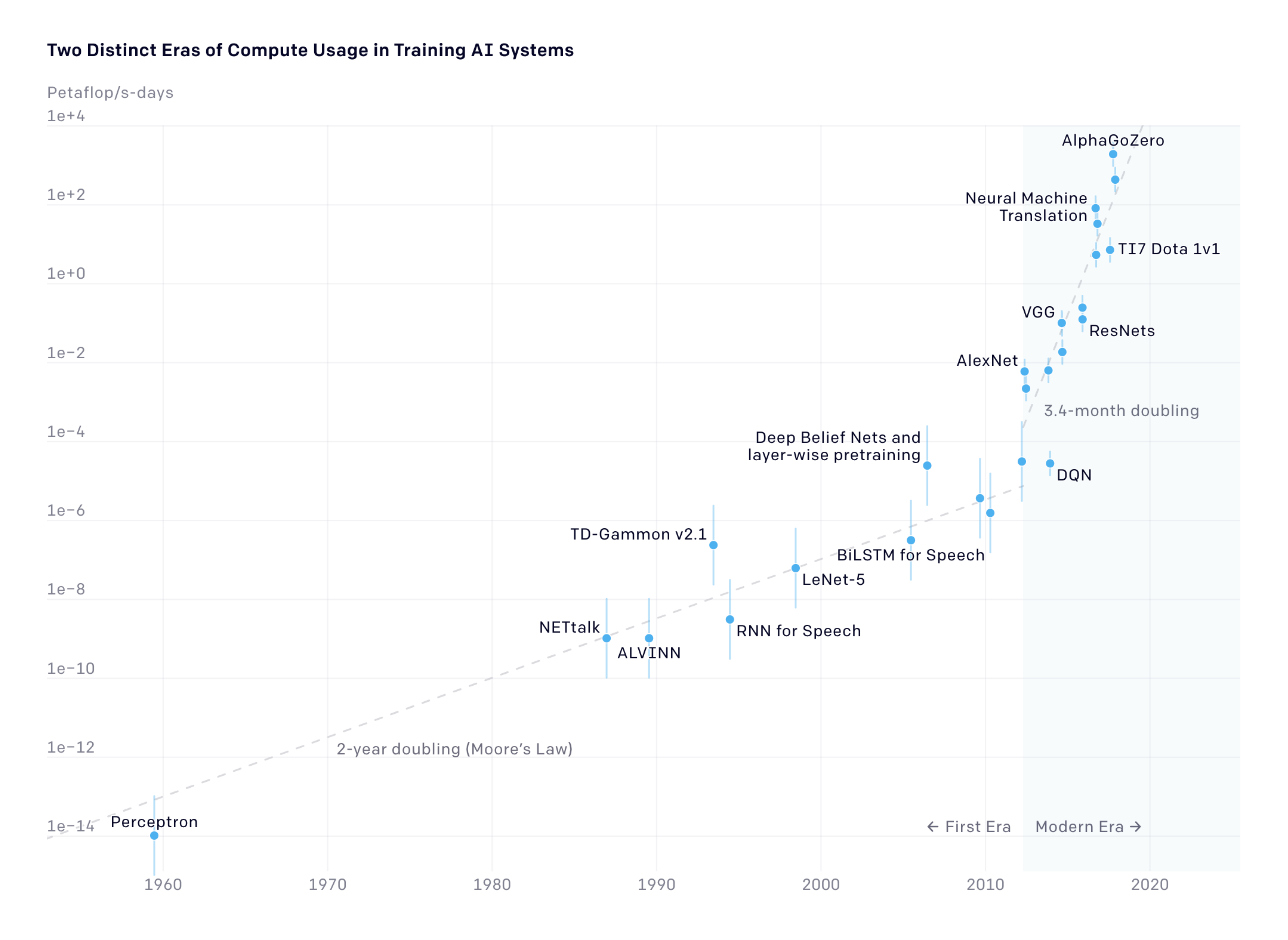


Figure 1 Language modeling performance improves smoothly as we increase the model size, datasetset size, and amount of compute[2] used for training. For optimal performance all three factors must be scaled up in tandem. Empirical performance has a power-law relationship with each individual factor when not bottlenecked by the other two.

scaling hypothesis




Two Distinct Eras of Compute Usage in Training AI Systems
Petaflop/s-days
1e+4
1e+2
1e+0
1e−2
1e−4
1e−6
1e−8
1e−10
1e−12
1e−14
1960
1970
1980
1990
2000
2010
2020
AlphaGoZero
Neural Machine Translation
TI7 Dota 1v1
VGG
ResNets
AlexNet
3.4-month doubling
Deep Belief Nets and layer-wise pretraining
DQN
TD-Gammon v2.1
BiLSTM for Speech
LeNet-5
NETtalk
ALVINN
RNN for Speech
2-year doubling (Moore's Law)
Perceptron
← First Era
Modern Era →

more compute

- specialized processors/edge ai
- ram/bandwidth
- hpc/systolic designs

more data

- higher resolution
- number of samples
- better annotations
- semi-supervised

Component	Raw Size	Weight	Epochs	Effective Size	Mean Document Size
Pile-CC	227.12 GiB	18.11%	1.0	227.12 GiB	4.33 KiB
PubMed Central	90.27 GiB	14.40%	2.0	180.55 GiB	30.55 KiB
Books3†	100.96 GiB	12.07%	1.5	151.44 GiB	538.36 KiB
OpenWebText2	62.77 GiB	10.01%	2.0	125.54 GiB	3.85 KiB
ArXiv	56.21 GiB	8.96%	2.0	112.42 GiB	46.61 KiB
Github	95.16 GiB	7.59%	1.0	95.16 GiB	5.25 KiB
FreeLaw	51.15 GiB	6.12%	1.5	76.73 GiB	15.06 KiB
Stack Exchange	32.20 GiB	5.13%	2.0	64.39 GiB	2.16 KiB
USPTO Backgrounds	22.90 GiB	3.65%	2.0	45.81 GiB	4.08 KiB
PubMed Abstracts	19.26 GiB	3.07%	2.0	38.53 GiB	1.30 KiB
Gutenberg (PG-19)†	10.88 GiB	2.17%	2.5	27.19 GiB	398.73 KiB
OpenSubtitles†	12.98 GiB	1.55%	1.5	19.47 GiB	30.48 KiB
Wikipedia (en)†	6.38 GiB	1.53%	3.0	19.13 GiB	1.11 KiB
DM Mathematics†	7.75 GiB	1.24%	2.0	15.49 GiB	8.00 KiB
Ubuntu IRC	5.52 GiB	0.88%	2.0	11.03 GiB	545.48 KiB
BookCorpus2	6.30 GiB	0.75%	1.5	9.45 GiB	369.87 KiB
EuroParl†	4.59 GiB	0.73%	2.0	9.17 GiB	68.87 KiB
HackerNews	3.90 GiB	0.62%	2.0	7.80 GiB	4.92 KiB
YoutubeSubtitles	3.73 GiB	0.60%	2.0	7.47 GiB	22.55 KiB
PhilPapers	2.38 GiB	0.38%	2.0	4.76 GiB	73.37 KiB
NIH ExPorter	1.89 GiB	0.30%	2.0	3.79 GiB	2.11 KiB
Enron Emails†	0.88 GiB	0.14%	2.0	1.76 GiB	1.78 KiB
The Pile	**825.18 GiB**			**1254.20 GiB**	**5.91 KiB**

Table 1: Overview of datasets in the Pile before creating the held out sets. Raw Size is the size before any up- or down-sampling. Weight is the percentage of bytes in the final dataset occupied by each dataset. Epochs is the number of passes over each constituent dataset during a full epoch over the Pile. Effective Size is the approximate number of bytes in the Pile occupied by each dataset. Datasets marked with a † are used with minimal preprocessing from prior work.

new approaches

- autodiff/backprop
- predictive coding
- jax/deepspeed

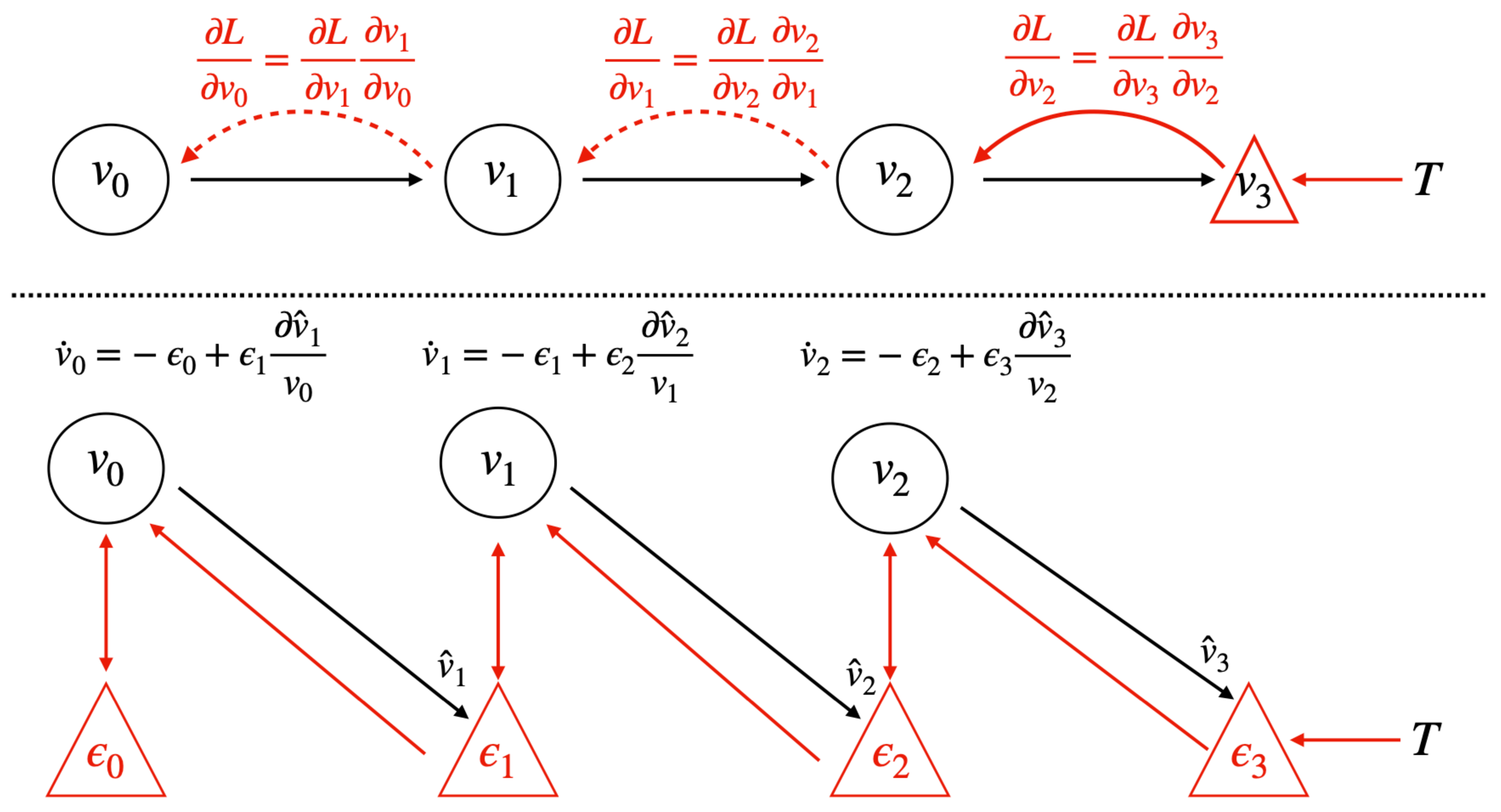


Figure 1: Top: Backpropagation on a chain. Backprop proceeds backwawrds sequentially and explicitly computes the gradient at each step on the chain. Bottom: Predictive coding on a chain. Predictions, and prediction errors are updated in parallel using only local information.

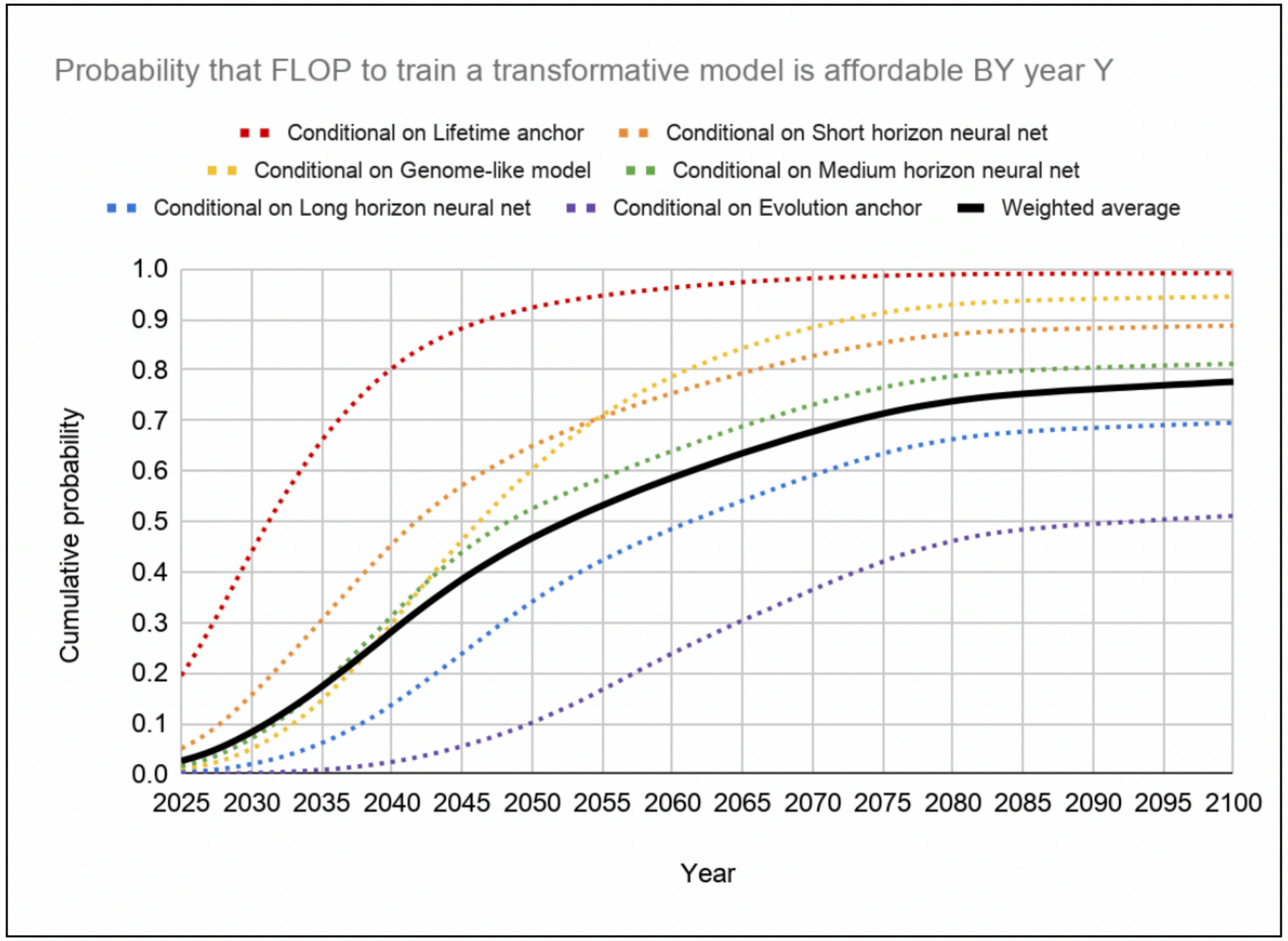
ai winter redux

- bitcoin / crypto
- commercial applications (cv, nlp, recommendation systems)
- autonomous devices
- above --> $$ --> more compute/data/r+d

general ai (agi)

- what is intelligence
- are humans special
- specialization vs generalization
- open world problem

projections



The black curve depicts the probabilities output by the weighted combination of hypotheses; the colored curves correspond to what the probability would be if we conditioned on a particular hypothesis. As discussed in the previous sections, the probabilities in the latter half of the century are likely slightly too low,

future is you

- ten years ago <--> ten years ahead
- never been a better time to get started
- fast.ai, gpu, find a problem
- eleuther.ai