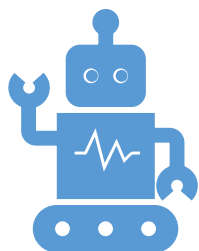




UiO : **University of Oslo**

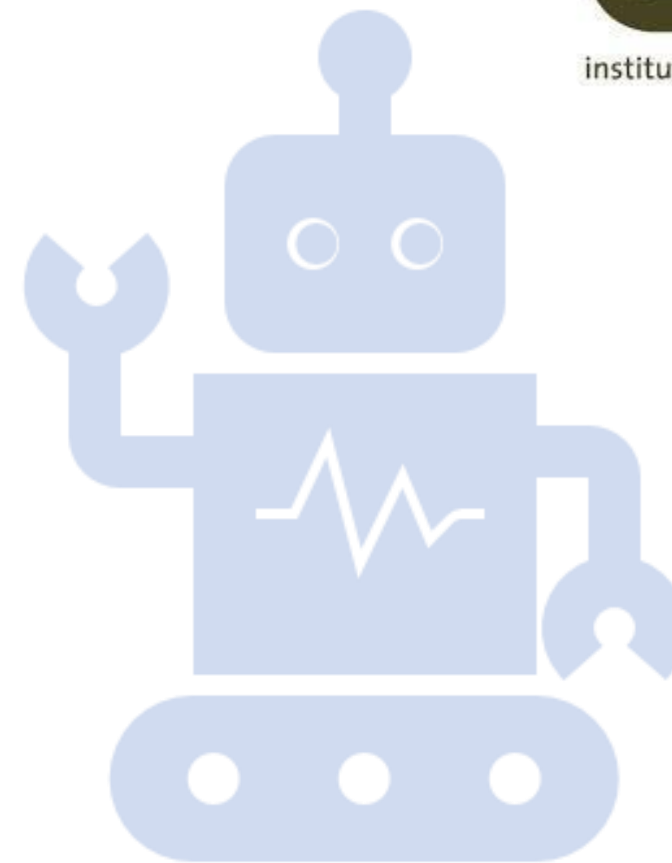


IN3050/IN4050 - Introduction to Artificial Intelligence and Machine Learning

Lecture 9

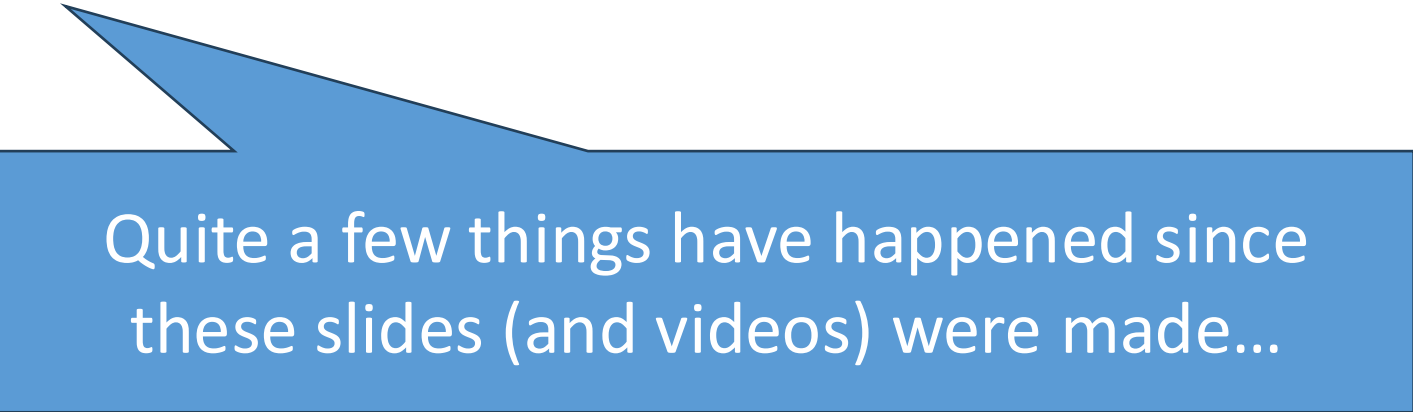
Glimpses beyond: Deep Neural Networks

Yves Scherrer (slides adapted from Jan Tore Lønning)



Today

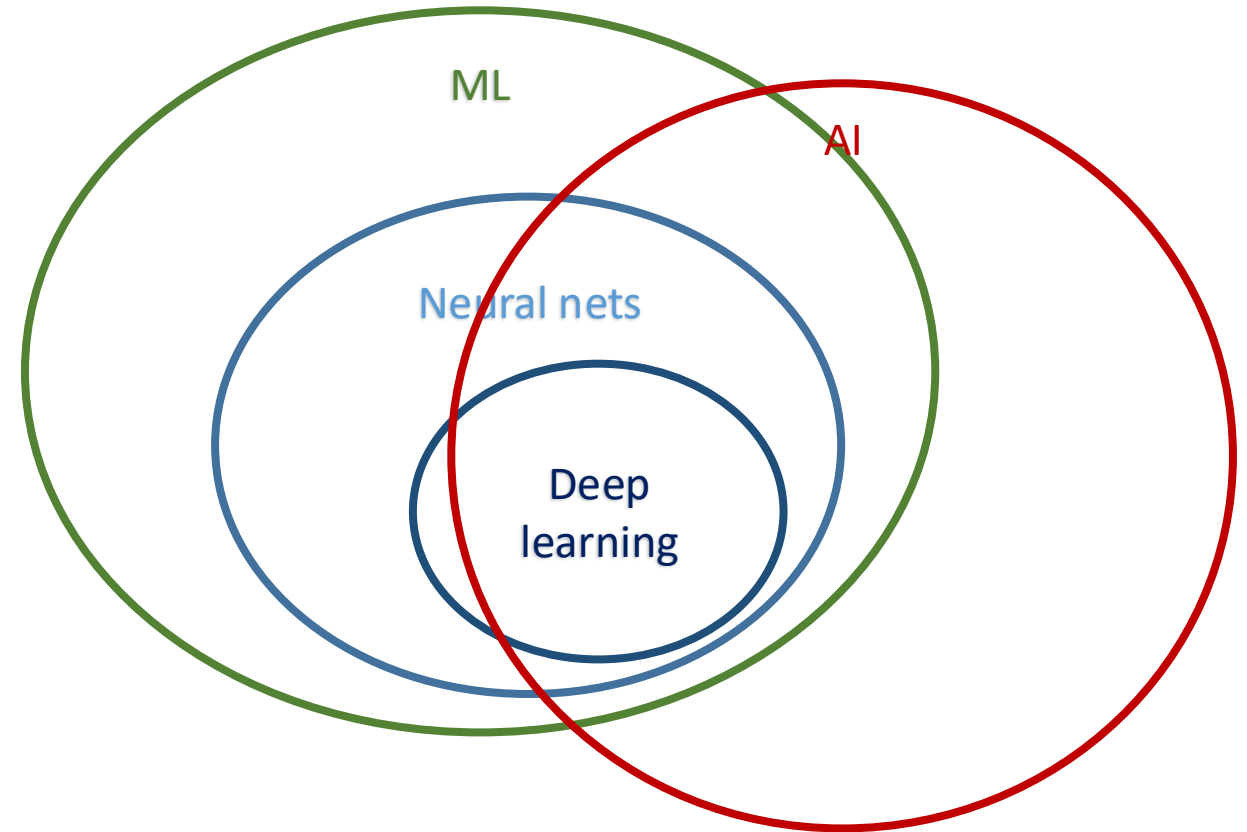
1. The deep learning revolution
2. Deep feed-forward neural networks
3. Convolutional NNs and image processing
4. Recurrent NNs and language processing



Quite a few things have happened since these slides (and videos) were made...

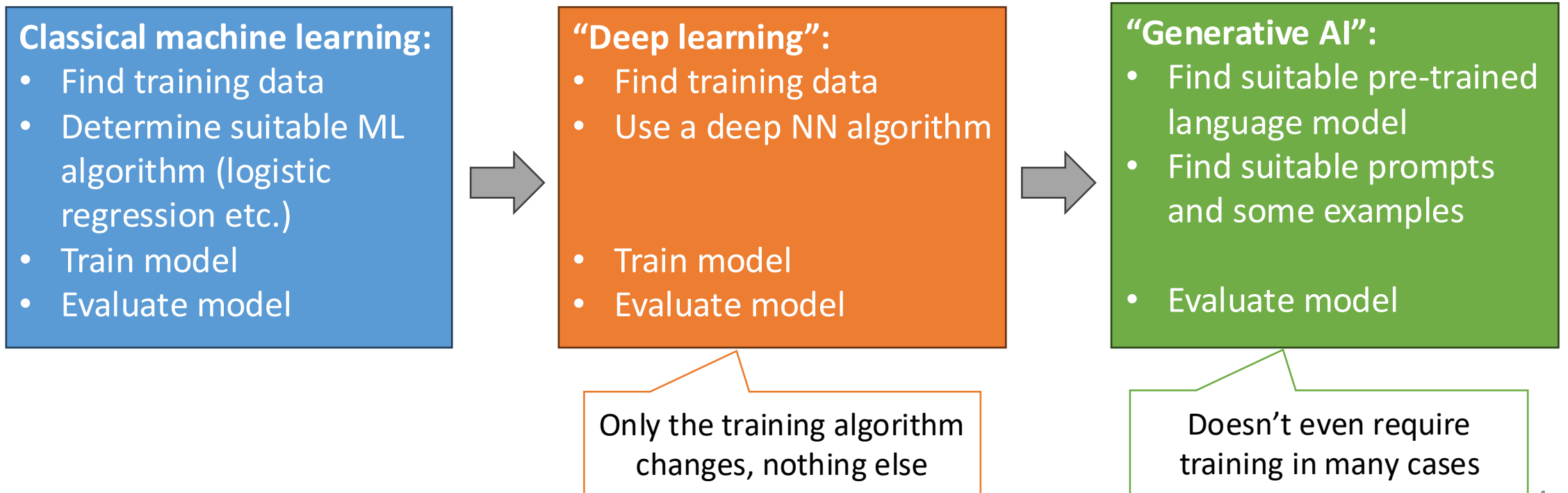
Deep learning

- A sub-class of neural nets
- No exact definition:
 - More an attitude/approach than a defined class
 - Normally: at least two hidden layers
 - Often: a much more specific architecture



Deep learning and AI

- Deep learning was a buzzword in its early days (2011 – 2019)
- Nowadays, people talk less about deep learning, but more about AI

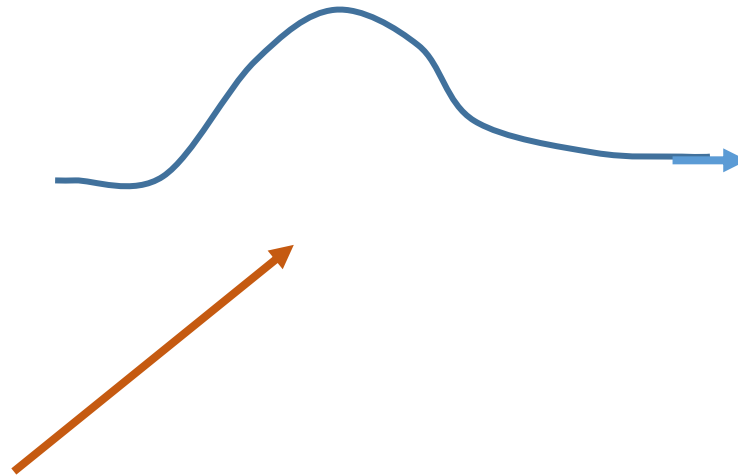


History of neural networks

Three main epochs:

1. The beginning (→ 1969)
2. Backpropagation (1986-)
3. Deep learning (2011→)

- Marsland, originally 2009, lacks (3)



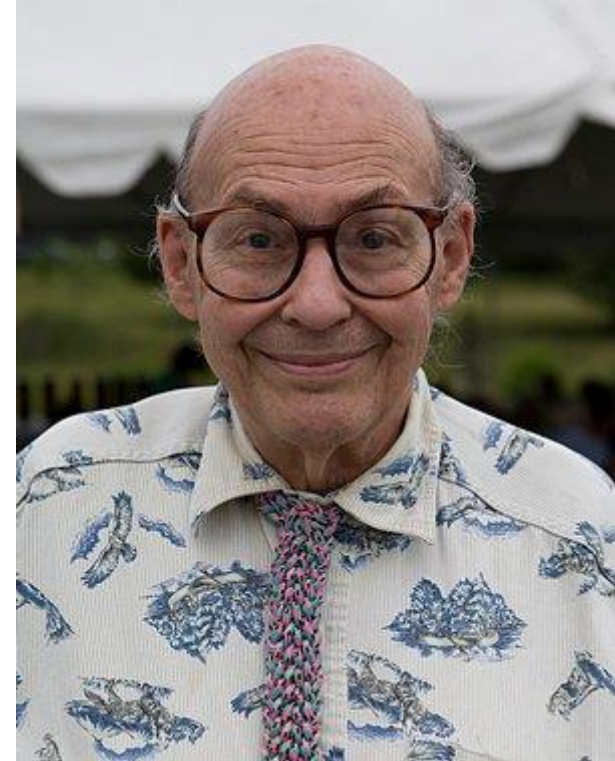
AI hypes and winters



Where do we stand today?
Will there be another AI winter?

NN.1: The beginning (→ 1969)

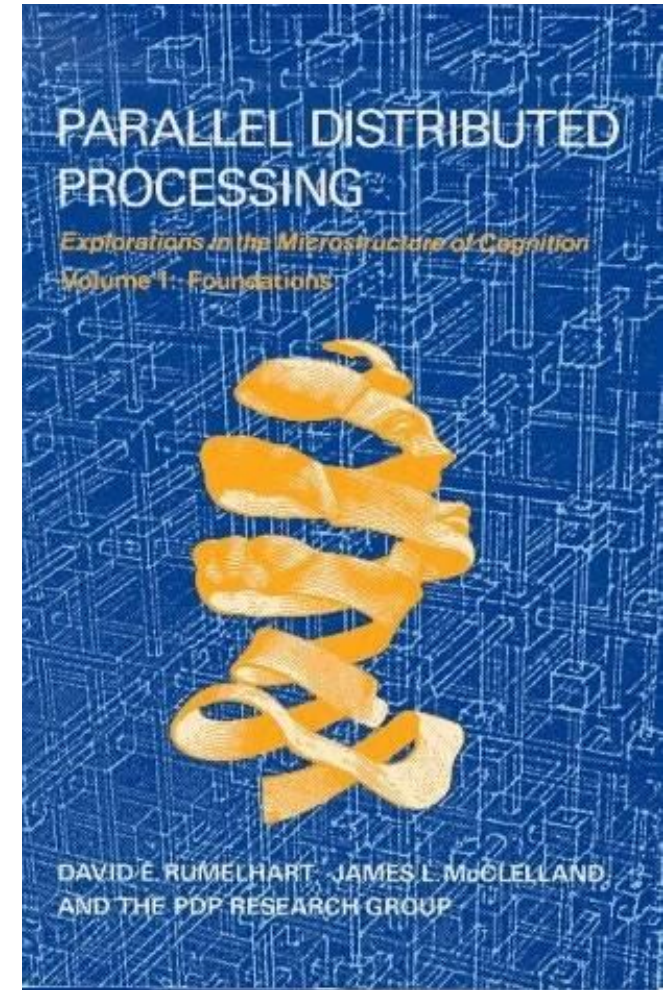
- 1958, **Rosenblatt** invented the perceptron
- 1969, **Minsky & Papert**, *The perceptron*:
 - Networks without hidden layers can only learn linear classifiers
 - Networks with hidden layers are probably impossible to train
- Less interest in perceptrons afterwards



Marvin Minsky (1927-2016)
AI pioneer, MIT AI Lab

NN.2: Backpropagation (1986-)

- 1986, **Rumelhart, Hinton, Williams** (re)invented backpropagation
- An immediate enormous interest by researchers
- But the practical results weren't impressing, and the interest diminished



NN.3: Deep learning

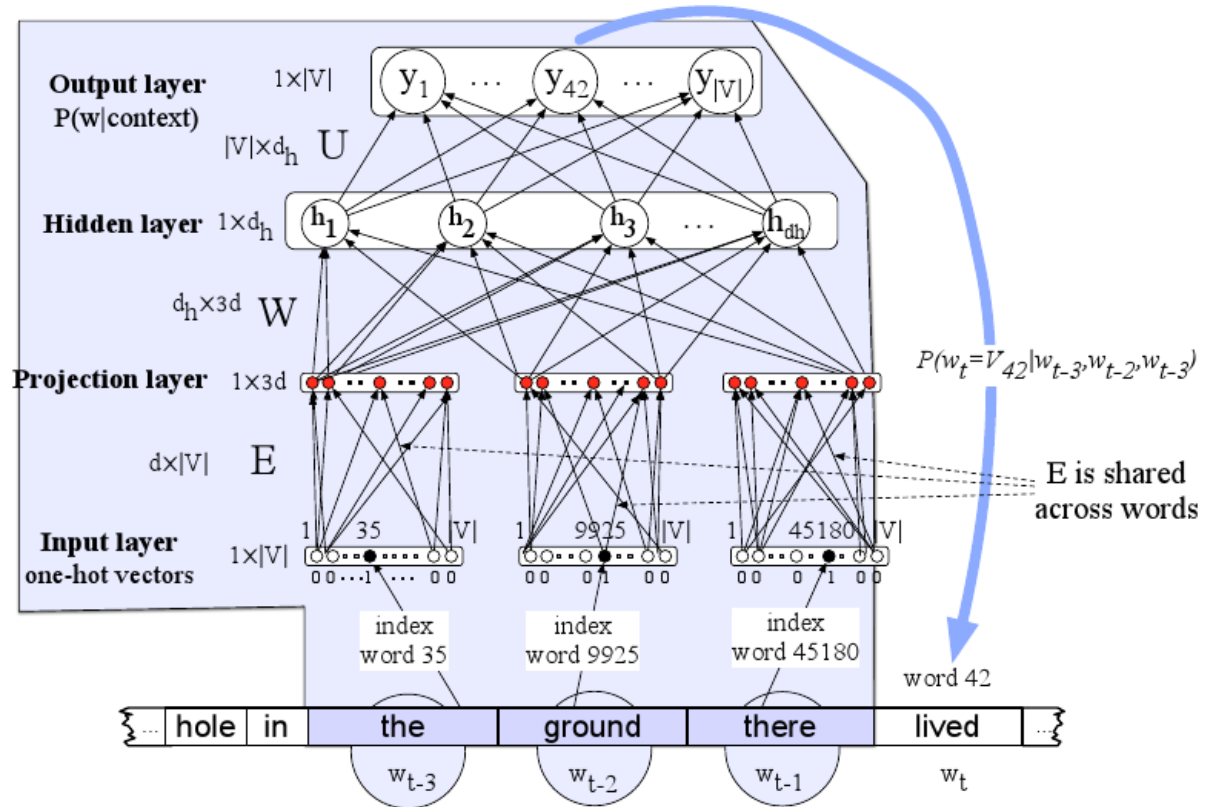
Note the name here! Nobody dared to call this AI at that time...

- In the 1990s and 2000s other approaches to ML were preferred in usage and developed, e.g., logistic regression, SVM
- But some brave and stubborn researchers continued the work on neural nets, including Hinton, and got their rewards in the 2010s



AI hype

- Today, most machine learning is done with multi-layered (“deep”) neural networks
- Bengio et al. (2003) proposed the first really working neural language model:
 - Feed-forward neural network architecture



AI hype

- In 2018, Bengio, Hinton and LeCun received the ACM A.M. Turing Award ‘for conceptual and engineering breakthroughs that have made deep neural networks a critical component of computing’.
- This signalled the end to the long ‘AI winter’.
- <https://awards.acm.org/about/2018-turing>



AI hype

However...

- In 2023, the same Geoffrey Hinton left his Google job and focused on talking about the **dangers of AI**:
<https://edition.cnn.com/2023/05/01/tech/geoffrey-hinton-leaves-google-ai-fears/index.html>
- ...more on that in later lectures.

This was mostly because of recent advances in large language models as chatbots.

Why did NNs finally succeed?

- More powerful machines
 - GPUs (graphical processing units)
- More data
- Better models



Increased compute

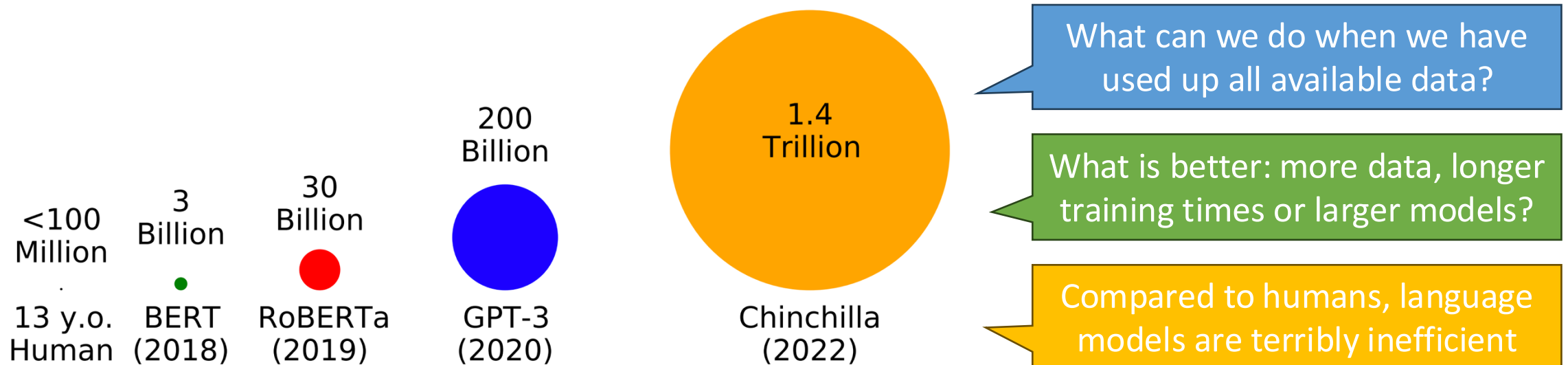
- Hardware capabilities are growing:
graphic processing units (GPUs) and **tensor processing units (TPUs)**.
They excel in parallelized matrix multiplication.
- But note compute divide: not everyone can afford burning 100K GPU/hours to train a GPT-7B model for a mid-sized language.
- Example: the LUMI supercomputer
 - 5th most powerful supercomputer in the world, 1st in Europe (at opening in 2022)
 - 2978 compute nodes with AMD MI250X GPUs (24 000 GPUs in total)



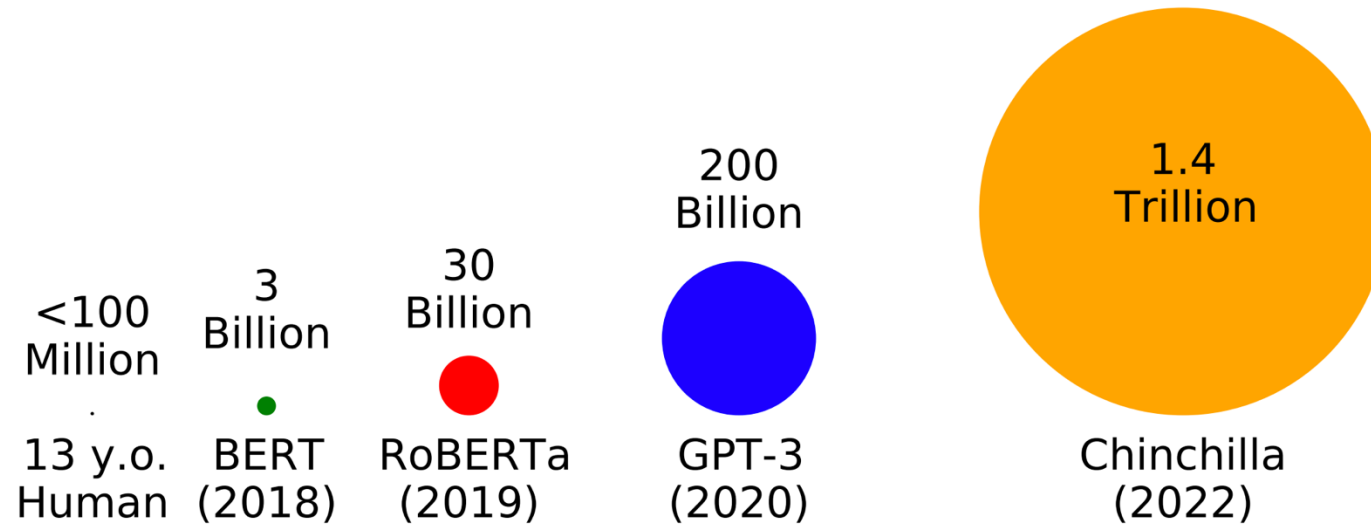
Increased data

Machine learning models are trained on large datasets:

- Language models mostly use data crawled from the Internet
- Most of this data is in English
- Even if large, the amount of usable data from the Internet is finite



Increased data



- What about ChatGPT?
 - Size of the training data unknown (but a mix of texts and code)
- Not all languages are equal in the size of available data
 - For Norwegian: not more than 50 billion words publicly available

Better models: *'The times they are a-changin'*

2009:

There are two things that we can do: add some backwards connections, so that the output neurons connect to the inputs again, or add more neurons. The first leads into recurrent networks. These have been studied, but are not commonly used. (Marsland, p. 71)

2017:

- Today, recurrent networks (RNNs) are heavily used, in Language Technology

2021:

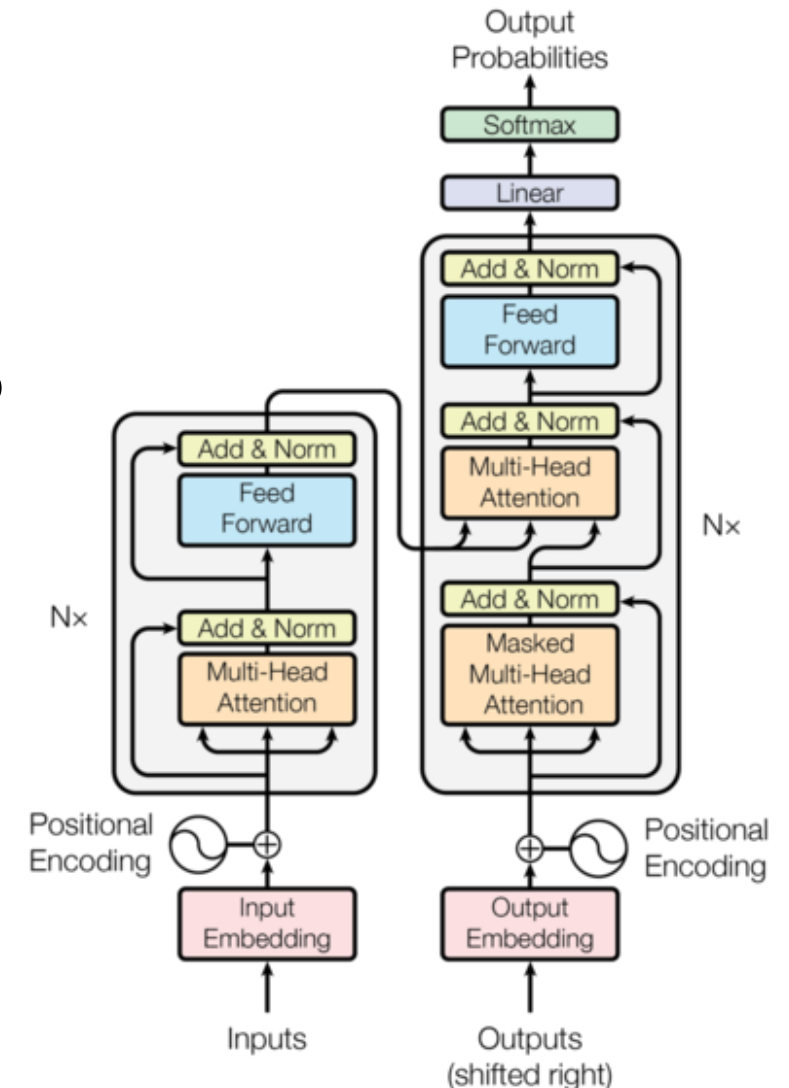
- But they are already giving way to the Transformer architecture

2025:

RNNs have mostly disappeared from recent research

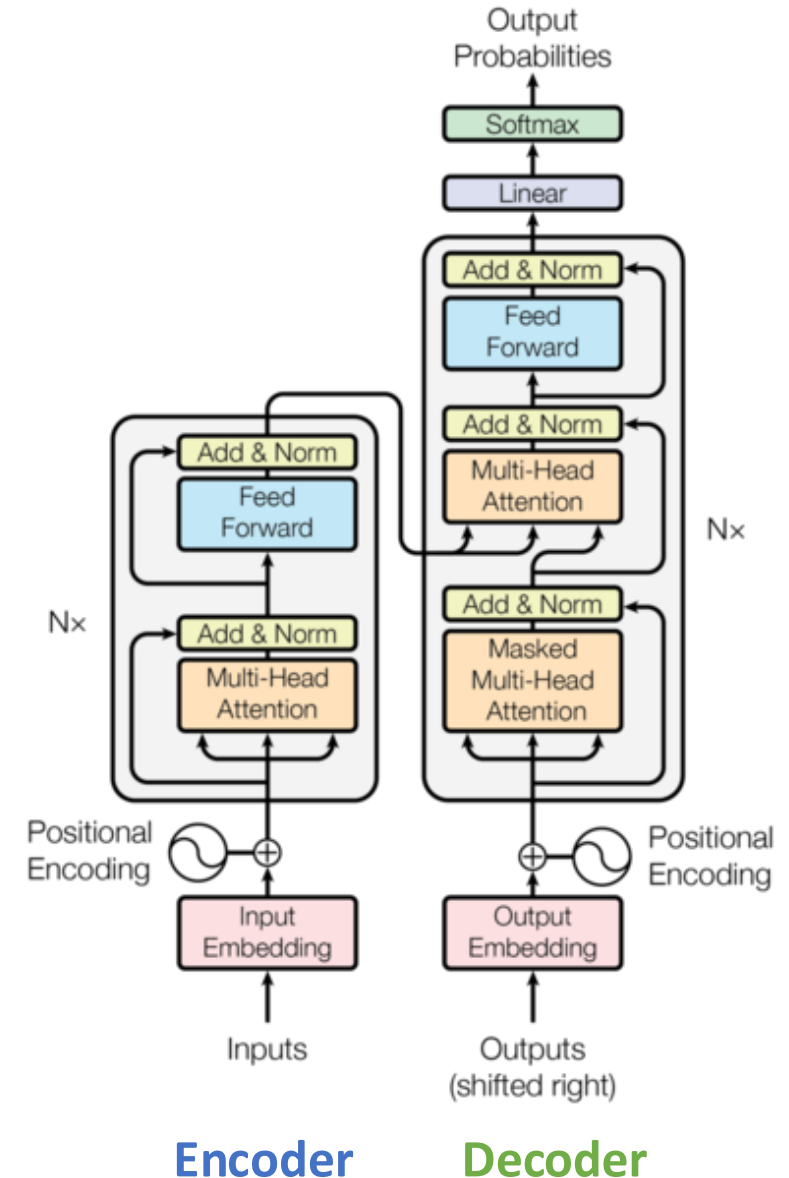
Better models: Transformers

- A sequence of feed-forward layers with **multi-headed self-attention**
 - The model learns what elements in the input sequence to pay attention to during training
 - All input elements are processed simultaneously
 - Training is easily **parallelized** across multiple computation units (unlike RNNs)
 - Many heads: solves the under-parameterization problem, different heads excel in different tasks
- Transformers made it possible to use the existing data and compute “in the most optimal way”
 - At least according to today’s knowledge...

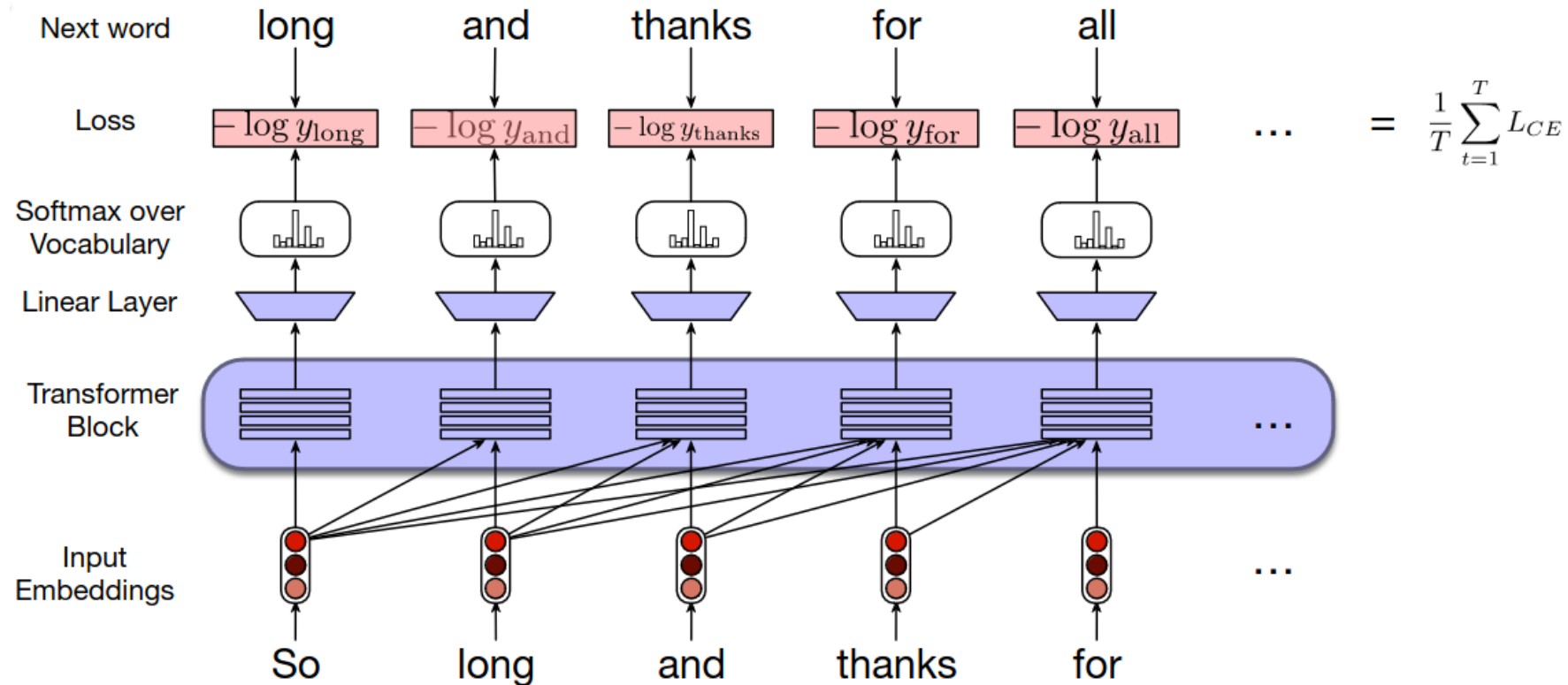


Better models: Transformers

- The original **Transformer** is composed of an **encoder** and a **decoder**
- It was initially developed for sequence-to-sequence transformation tasks like machine translation
 - **Encoder**: source language
 - **Decoder**: target language
- But its building blocks are used in almost all LLMs:
 - **BERT** models only contain an **encoder**
 - **GPT** models only contain a **decoder**



Transformers as language models



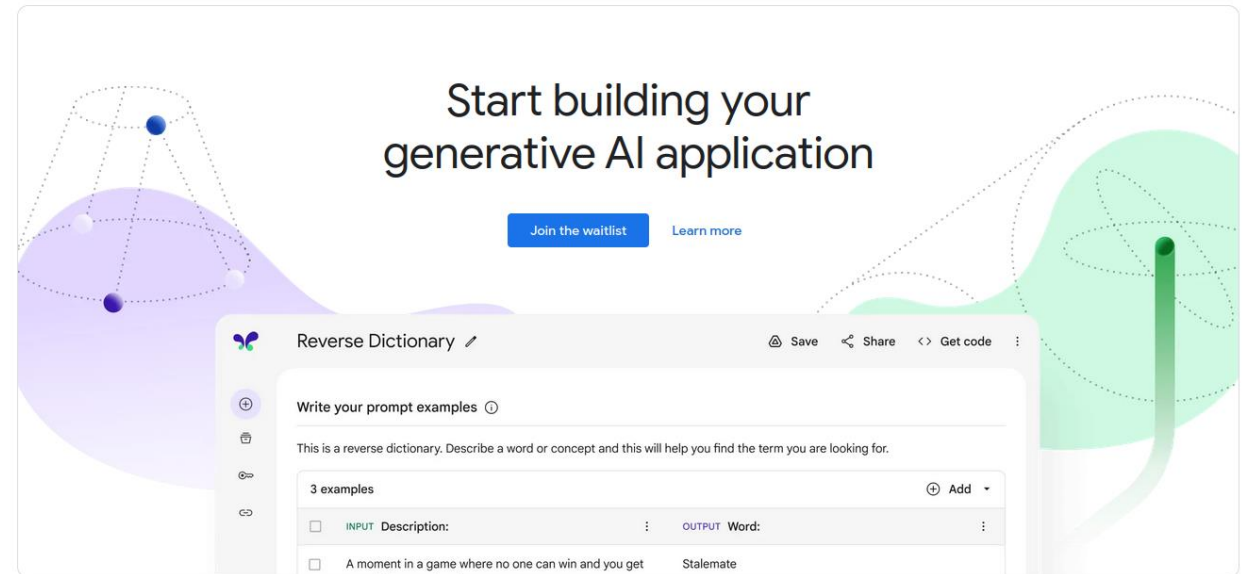
Learn more about Transformers in IN4080 and IN5550!

Language models

Language models are trained to predict the next word



ChatGPT, a generative language model by OpenAI



PaLM, a generative language model by Google

Language models

Any language model is a **text generator** by definition.

Autoregressive or **causal** generation:

- feed a word or a sentence (prompt) into the LM
- get a probability distribution over what words are likely to come next
- pick the most probable word from this distribution (or use some form of sampling)
- feed it right back in the LM together with the previous words
- repeat this process and you're generating text!

This is what ChatGPT or GPT-4 do. Thus, **generative language models**.

Text generation is not the only task LMs can do, but it pushed them into the headlines.

Language models

- Many of the popular LLMs are closed and only available to the public as black-box services.
- We have trained an open chat LLM for Norwegian: NorMistral
- Register for free and try it out:
<https://chat.llm.sigma2.no/>