Turing Machine and Deep Learning
Lecture 5: Recap + Frontiers

Satchit Chatterji
satchit.chatterji@gmail.com

MSc Artificial Intelligence
University of Amsterdam

9th June 2023

Recap of the Course

Lecture 1: Introduction

- PDS Recap
- Intro to ML Domains
- Statistical Learning Theory Basics + ML Components
- Linear + Polynomial regression
- Cross-validation and model selection
- Over/underfitting

🐢

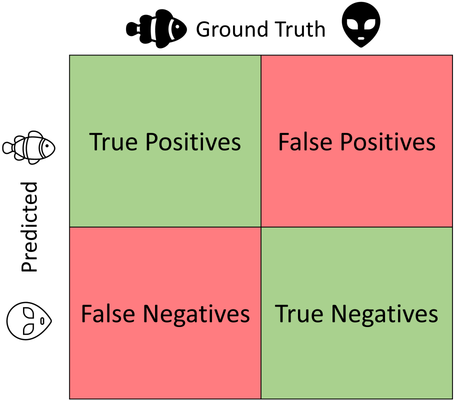
Such compressed poems
with seventeen syllables
can't have much meaning

🧔

Meaning lies as much
in the mind of the reader
as in the haiku.

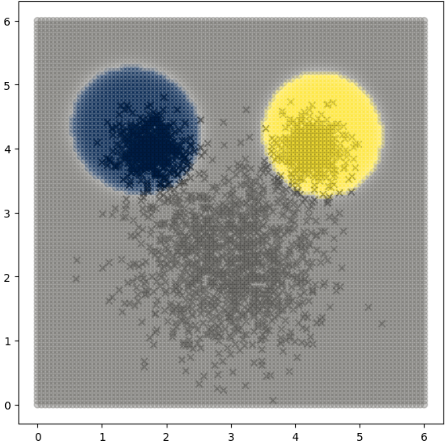
Lecture 2: Supervised ML

- Logistic regression
- Vectorization
- Classification metrics
- Confusion matrices
- Decision trees
- Random Forests
- SVMs



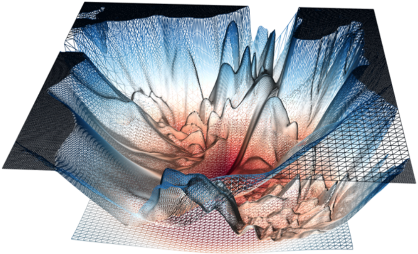
Lecture 3: Unsupervised ML

- Principal component analysis
- K-means clustering
- EM Algorithm
- Gaussian Mixture Models



Lecture 4: Neural Nets

- Regualization
- Building NN intuition
- Gradient descent
- Loss functions
- Universal approximation theorem
- Convolutions
- MLPs, CNNs RNNs
- NN Training pipeline
- k-fold Cross-Validation
- Training curves
- GPU Parallelization



Lecture 5: This!

- Recap of course
- Neurosymbolic integration introduction
- LLMs+ChatGPT introduction

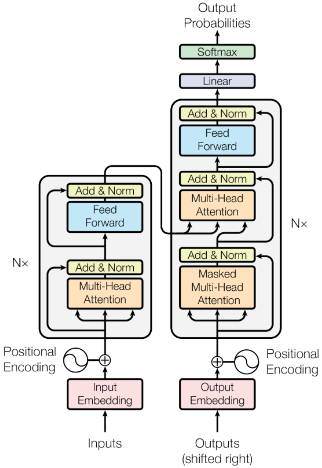


Figure 1: The Transformer - model architecture.

Neurosymbolic Integration

The Issue with Good Neural Networks

Figure: Dall-E 2: *An astronaut riding a horse in photorealistic style.*

Shakey (1984)

Figure: Shakey the Robot

```
UNBLOCK(DX,RX,BX)

    Preconditions:

        BLOCKED(DX,RX,BX) ∧ INROOM(ROBOT,RX) ∧ PUSHABLE(BX)

    Delete List:

        AT(ROBOT,$1,$2)
        BLOCKED(DX,RX,BX)
        AT(BX,$1,$2)
        NEXTTO(ROBOT,$1)
        NEXTTO(BX,$1)
        NEXTTO($1,BX)

    Add List:

        *UNBLOCKED(DX,RX)
         NEXTTO(ROBOT,BX)

Unblocks door DX by pushing object BX away from its place in room RX directly in
front of door DX.
```

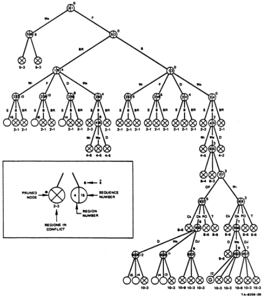


Figure 5: THE ANALYSIS TREE

Figure: Shakey the Robot

GOFAI

Good old-fashioned AI focussed on **symbolic reasoning**: knowledge is represented using symbols, can can be manipulated using formal rules and logical operations.

- Domain Knowledge: Expertly curated and represented information
- Rule Based Systems: Handcrafted reasoning
- Logic and Deductive Reasoning: Emulate human-like reasoning via formal logic
- Interpretability: We know exactly how a conclusion is reached

GOFAI vs Connectionism

Limitations of GOFAI:

1. Brittleness to high complexity
2. Knowledge engineering was (is) tough+expensive
3. Lack of learning ability
4. ...

Up and coming: *connectionism*: modelling learning through neural networks.

Connectionism

The new kid on the block...

- **First neurons**: McColloch & Pitts (1943)
- **The Perceptron**: Rosenblatt (1958)
- **Parallel Distributed Processing**: Rumelhart & McClelland (1986)
- **Backpropagation**: Rumelhart, Hinton, & Williams (1986)

Connectionism: The New Ruler?

Limitations of GOFAI:

1. Brittleness to high complexity
2. Knowledge engineering was (is) tough+expensive
3. Lack of learning ability
4. ...

Connectionism: The New Ruler?

Limitations of GOFAI:

1. Brittleness to high complexity
2. Knowledge engineering was (is) tough+expensive
3. Lack of learning ability
4. ...

Advantages of NNs:

1. Flexibility to complex problems
2. Just high-level representation choices needed
3. All about learning from data!
4. ...

GOFAI: The True Champion?

Advantages of GOFAI:

1. Interpretability and explainability is the core of system
2. Knowledge and reasoning is symbolic and human-like
3. No data needed – just rules for pure symbolic manipulation
4. ...

Limitations of Connectionism:

1. Severe lack of interpretability and explainability
2. Difficulty in capturing complex symbolic knowledge
3. Heavy reliance on large amounts of training data
4. ...

Neurosymbolic Integration

A new field that *could* show promise in combining neural networks and GOFAI.

- Integration of Symbolic Reasoning and Machine Learning
- Knowledge Representation
- Impact on Cognitive Science
- Explainable and Interpretable AI
- Combining Logic and Neural Networks

What is up with LLMs?

Overview

1. What are *language models*?
2. n-Gram models
3. Neural language modelling
4.

Language models

Language models (in ML) are **statistical models** that capture the probability distribution of sequences of words in a language.

They are designed to **estimate the likelihood of a word given its context**, whether it's the previous words in the sequence or the surrounding context in a document.

n-Gram models

- *Goal*: Estimate the probability of a word based on the previous $n-1$ words in the sequence.
- Example:

  James tells his story with great...

- Bi-gram ($n=2$) model – estimated probability from counts in a training corpus:

$$P(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n)}{\sum_{w^*} C(w_{n-1}w^*)} \approx \frac{C(w_{n-1}w_n)}{C(w^*)}$$

- Probability of a sequence of words:

$$P(W_1^n) \approx \prod_{k=1}^{n} P(w_k|w_{k-1})$$

n-Gram generation

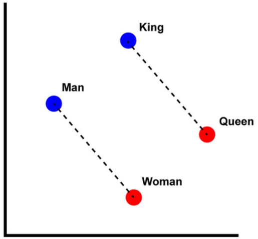
Some Shakespeare...

2 gram	–Why dost stand forth thy canopy, forsooth; he is this palpable hit the King Henry. Live king. Follow. –What means, sir. I confess she? then all sorts, he is trim, captain.
3 gram	–Fly, and will rid me these news of price. Therefore the sadness of parting, as they say, 'tis done. –This shall forbid it should be branded, if renown made it empty.
4 gram	–King Henry. What! I will go seek the traitor Gloucester. Exeunt some of the watch. A great banquet serv'd in; –It cannot be but so.

* Sourced from `https://cl-illc.github.io/nlp1-2022/resources/slides/NLP1-lecture2.pdf`

Neural Language Models

- *Goal*: leverage deep learning techniques to learn word representations and capture contextual dependencies.
- *Example*: **word2vec** (Mikolov et al, 2013)
  - Shallow, two-layer NN, which aims to predict the surrounding words given a target word or predict a target word given its context.
  - Captures interesting properties of words, e.g. semantic similarities, analogies, etc



- From MLPs and RNNs, the field quickly moved to *transformer models*: **Attention Is All You Need** (Vaswani et al., 2017)
- Uses the 'attention mechanism' (Bahdanau et al, 2014).
- Captures long-range dependencies better than previous models

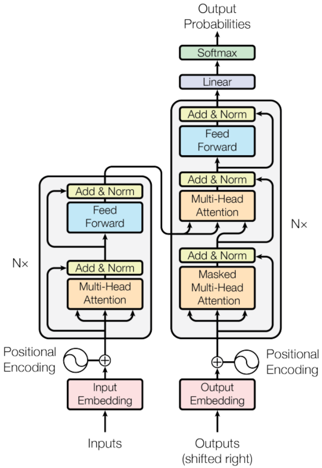
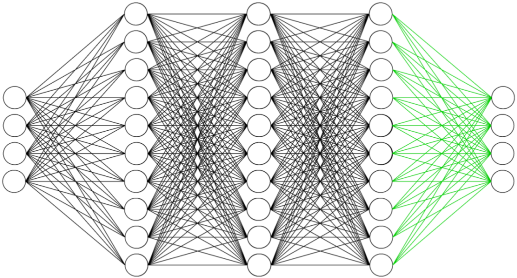


Figure 1: The Transformer - model architecture.

Tangent: Fine-tuning

Fine-tuning a model refers to training *some* of the layers of an already trained ('pretrained') neural network model, usually the final layers.



This can help in several situations: e.g. use a large, general dataset to learn general features about a set of inputs, then retrain the last layer(s) on a smaller, more specific dataset.

Back to Representations

The goal of models like word2vec, BERT (Devlin et al. 2018) and GPT (Radford et al, 2018) is to capture a dense, vector *representation* of words (a 'word-embedding') given the context in which they are present.

Generally, words that are semantically similar have similarly-directed vectors, that is, the cosine similarity is high:

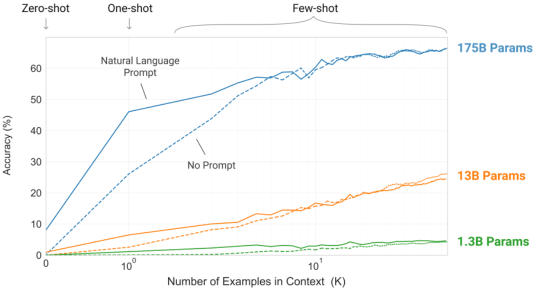
$$\text{cos\_sim}(v_1, v_2) = \frac{v_1 \cdot v_2}{||v_1|| \cdot ||v_2||}$$

This enables e.g. searching for words that match *semantically* and not just exact matches. This enables far better results in NLP tasks such as *sentiment analysis* or *topic modelling*, especially after fine-tuning.

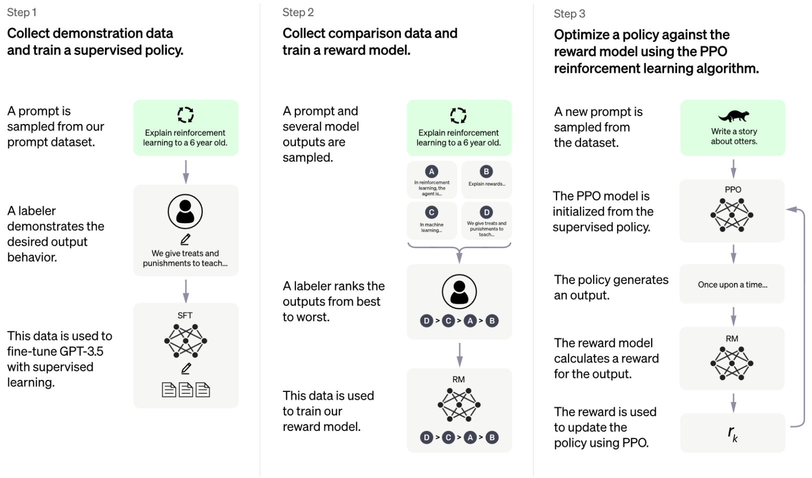
Towards ChatGPT: The GPT-3 Model

Idea: More data + larger models ⇒ better + more general performance? (ref. Sutton, 2019)

GPT: **Generative Pretrained Transformer**



ChatGPT Finetuning Pipeline



Source: `https://openai.com/blog/chatgpt`