Advanced neural network architectures

Dr. Charles Kind

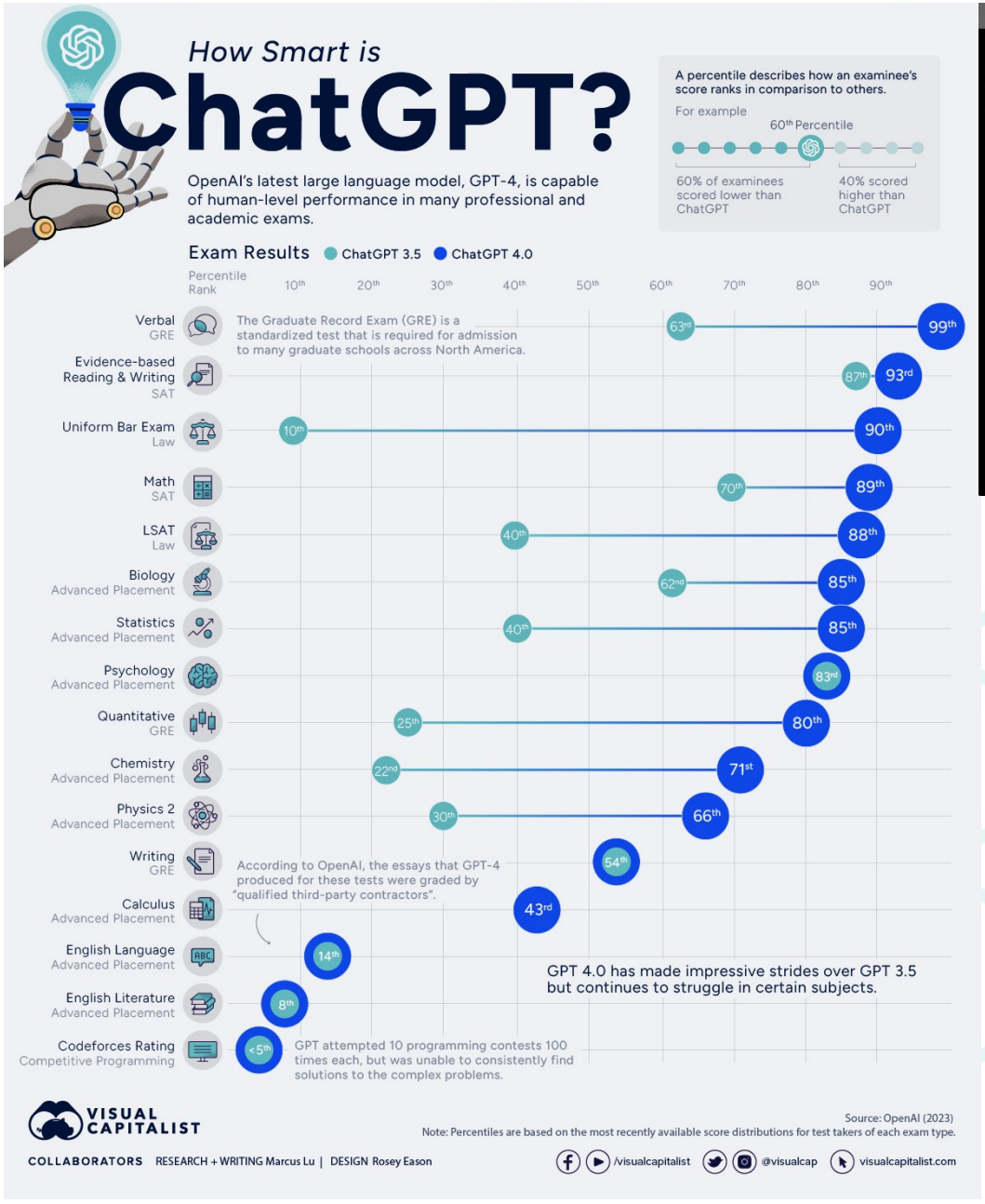
Bristol University

May 2023

Learning objectives

- What is natural language processing and what is ChatGPT.
- What are the complexities of language processing:
  - Why standard neural networks cannot process language.
  - Why convolutional neural networks cannot process language.
- What are pairwise CNN?
- How, and why, do we encode sentence order into a neural network representation.
- How do transformers work?
- Is their a human brain equivalent?

ChatGPT



Above: a 3rd party conversational tool to make ChatGPT's discourse more varied!

Left: An analysis of ChatGPT's ability to pass exams.

Clearly ChatGPT is an immensely powerful AI, capable of passing a Turing test. How does it work?

ChatGPT

- How does ChatGPT work?
- GPT stands for ‘generative pre-trained transformers’.
- In 2017 convolutional neural networks (CNN’s) achieved major breakthroughs in image processing.
- Could CNN’s be applied to natural language processing (NLP)?
- NO! CNN’s, in most cases, failed spectactularly.
- Why? What are the differences between text and images?
  - Text is not described by numbers.
  - An image pixel is directly related to it’s neighbours, this need not be the case with words.
  - Breaking text down using convolution implies that the relationship of words proximate to each other is how language is structured … but it’s not.

One-Hot encoding

- How do we encode language as numbers?
- List all distinct words in a language in a vector.
- The positions of those words becomes their encoding ie:

$$\begin{bmatrix} the \\ cat \\ sat \\ on \\ mat \\ \vdots \end{bmatrix} \Longrightarrow \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ \vdots \end{bmatrix} \Longrightarrow \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \dots \\ 0 & 1 & 0 & 0 & 0 & \dots \\ 0 & 0 & 1 & 0 & 0 & \dots \\ 0 & 0 & 0 & 1 & 0 & \dots \\ 0 & 0 & 0 & 0 & 1 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

- Hence ...

Mat sat on cat

$$\left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ \vdots \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ \vdots \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ \vdots \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ \vdots \end{bmatrix} \right.$$

Transformers

- Consider the sentence:
  ”the frightened cat was running so fast that it did not notice the thorn in it’s foot”
- The subject ‘cat’ and the objects ‘thorn’ and ‘foot’ a far away from each other.
- CNN’s cannot handle this as this distance can essentially be arbitrary.
- We shall try replacing CNN’s with what we are going to call a pairwise CNN.
- First we turn the sentence into an array of pairs of all the words.

$$\text{mat sat cat} \implies \begin{bmatrix} \text{mat mat} & \text{mat sat} & \text{mat cat} \\ \text{sat mat} & \text{sat sat} & \text{sat cat} \\ \text{cat mat} & \text{cat sat} & \text{cat cat} \end{bmatrix}$$

- Making pair vectors of sentences.

Transformers, pair and pair again

- Pairing means distance between words no longer matters.
- We can now use a CNN on each pair … but wait.
- Now we pair the pairs and train another layer of CNN's on the length four vectors!
- We keep doing this with all possible combinations hence … this is equivalent to training a standard CNN on all permutations of sentence ordering.
- The basic premise is that somewhere in the list of all combinations must be some whose order is 'good' in the sense that all related words appear optimally close to each other.
- Sorted … but not quite. Order does matter for example:

'then the bird flies' or 'flies then the bird'

Transformers, encode location

- We can encode our words along with their location in the sentence and then train.
- At last we have model that can work … hooray.
- But wait …
  - We know that combinatorics can generate large numbers
  - We are talking about breaking sentences down into all possible combinations including pairings and pairings of pairings etc
  - This model is HUGE and as sentence length increases it grows exponentially
- So what can we do? With image recognition CNN's we averaged could this work here? Could we average our column vectors at every step and reduce our work from n^2 to n?

Transformers, averaging pairs

- If we simply average all pairs of words (or pairs of pairs etc) we will lose a great deal of information. Imagine averaging a sequence of unrelated images.
- What we would ideally like is to average the pairs but give greater strength to those word pairs that are important.
- Consider:

  ”There was a food in the oil and it was spitting”

  “There was a cobra in the oil and it was spitting”

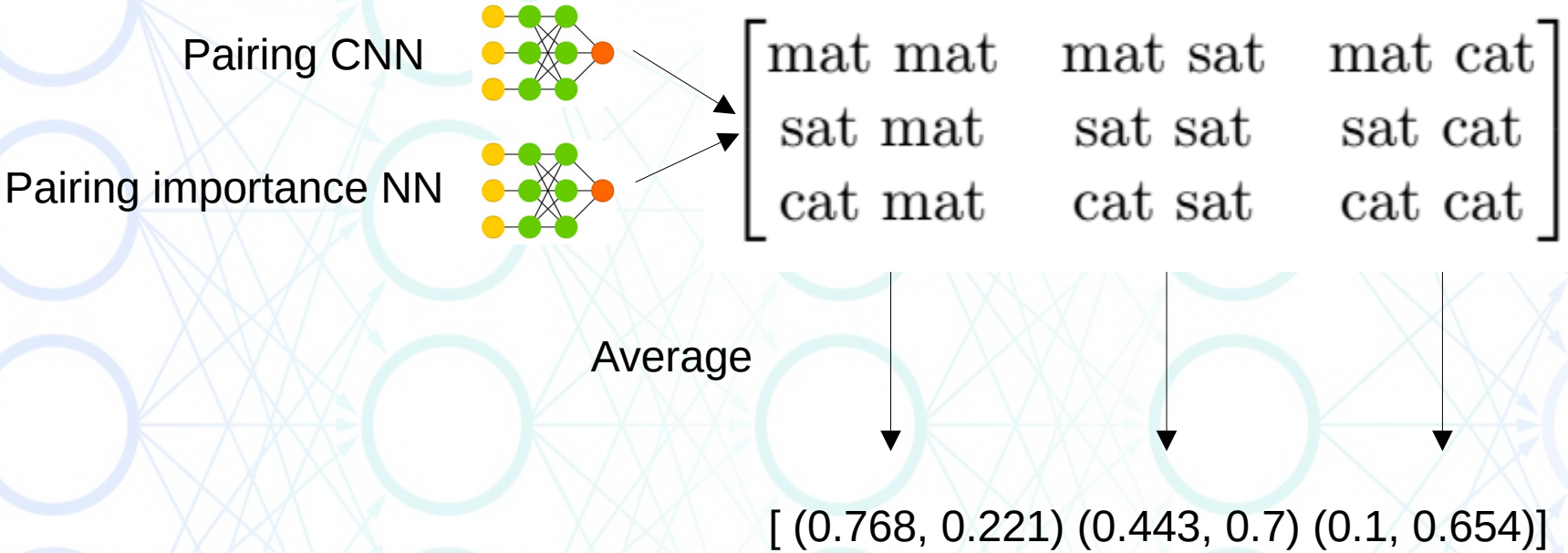
- In the first sentence the oil is spitting, in the second the cobra is doing the spitting.
- How do we decide which pairs are important before our averaging?

Transformers, averaging pairs

- We train another neural network to analyse the importance of pairings!
- This means we have two neural networks per layer.
  - The first maps pairs (pairs of pairs etc) to their new representations
  - The second maps pairs to their importance.
- We can now average our columns using the importance to weight the average.
- This entire layer is called a ‘self-attention layer’
- This then is the fundamental method by which transformers work.
- It can be (and is) heavily optimised to minimise calculations.

Transformers, averaging pairs

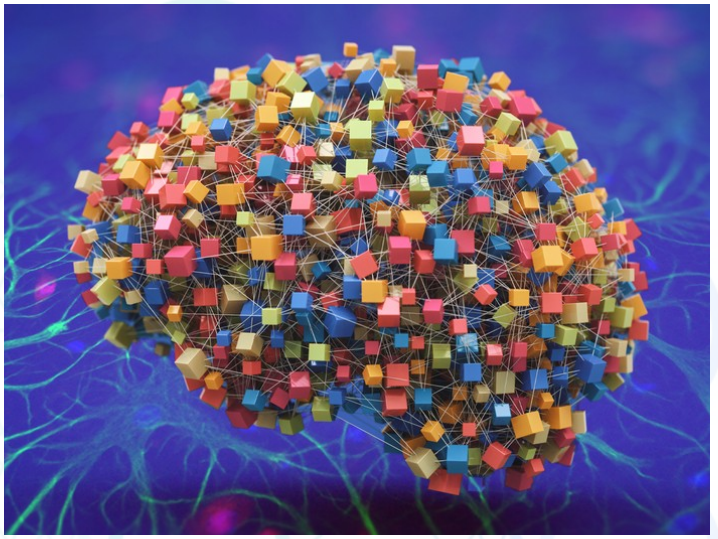
Self attention layer

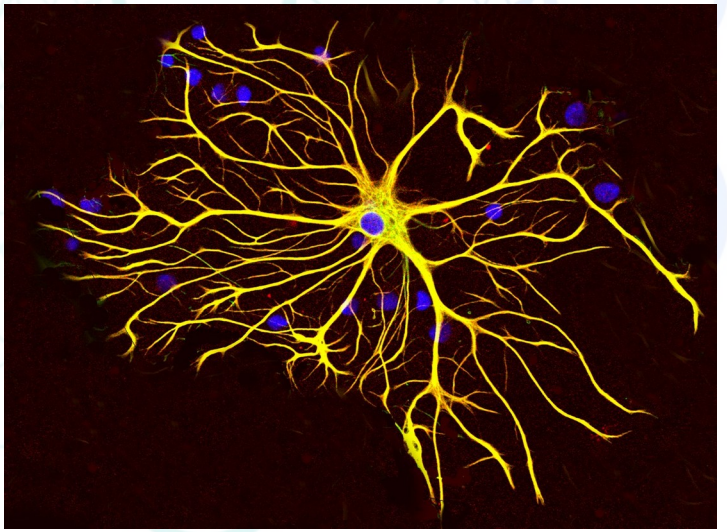


- We can optimise many of these operations. For example we can replace the pairing CNN with a linear function and recover our non-linearity by using a CNN after averaging.
- According to OpenAI, the training process of Chat GPT-3 required 3.2 million USD in computing resources alone.
- This cost was incurred from running the model on 285,000 processor cores and 10,000 graphics cards, equivalent to about 800 petaflops of processing power.
- It would take around 120 years to train GPT-3 on a single NVIDIA RTX 4090
- ChatGPT's training included a great deal of human input with people playing the roles of chat bot and human as well as people to label and weight harmful content.

Transformers, in your brain?

- But it seems absurd that such a mechanism exists in our brain, or does it?
- We saw that there where strong correlations between convolutional DNN and human image recognition.
- Sometimes an astrocyte is connected to a synapse creating a three part synapse.
- One astrocyte may form millions of tripartite synapses.
- The astrocyte collects some neurotransmitters that flow through the synaptic junction. At some point, the astrocyte can signal back to the neurons.
- Astrocytes operate on a much longer time scale than neurons … buffer memory.
- Using this addition to the neural model researchers have shown that neural networks in the human brain can act as transformers.
- See Kozachev et. al, ‘Building transformers from neurons and astrocytes’, PNAS, 2023





Astrocyte: a sub-type of glial cells

Transformers

- Imagine we want to train a translator say English → German.
- English sentences go through the input block.
- The German translations through the output block.
- The ‘Masked’ change to the attention block is a process whereby words are masked from the neural network and it tries to predict what they will be. This is how it learns.
- Words are paired between now languages and weighted.
- “I arrived at the bank after crossing the…” requires knowing if the sentence ends in “... road.” or “... river.”
- Vaswani et, al, ‘Attention Is All You Need’, arXiv, 2017