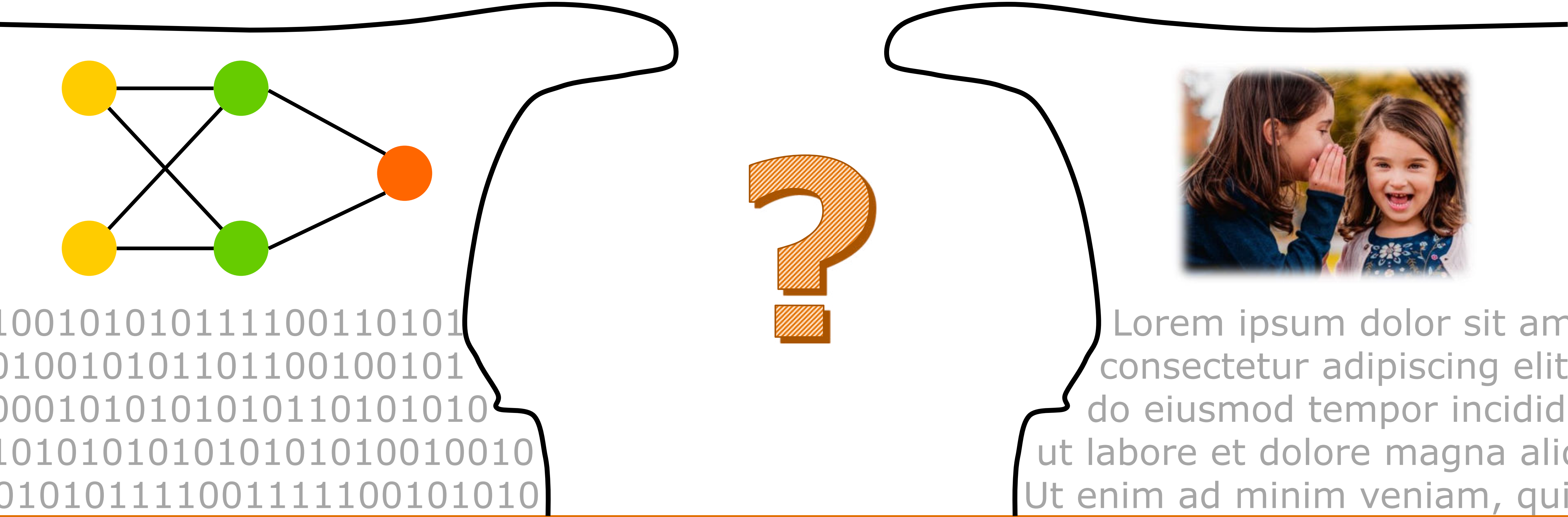
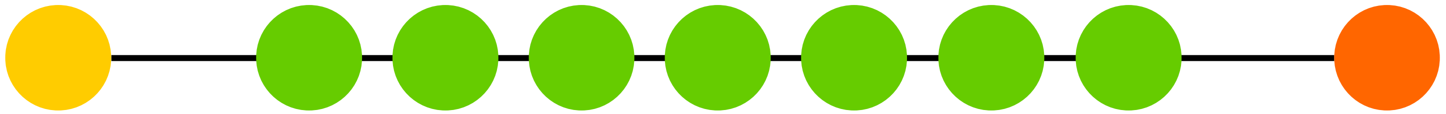
C. From ANNs towards LLMs

1. How does an ANN work with words?
2. Latest LLM developments
3. Hypotheses – role of information specialists
4. Plenary sessions 5 and 6

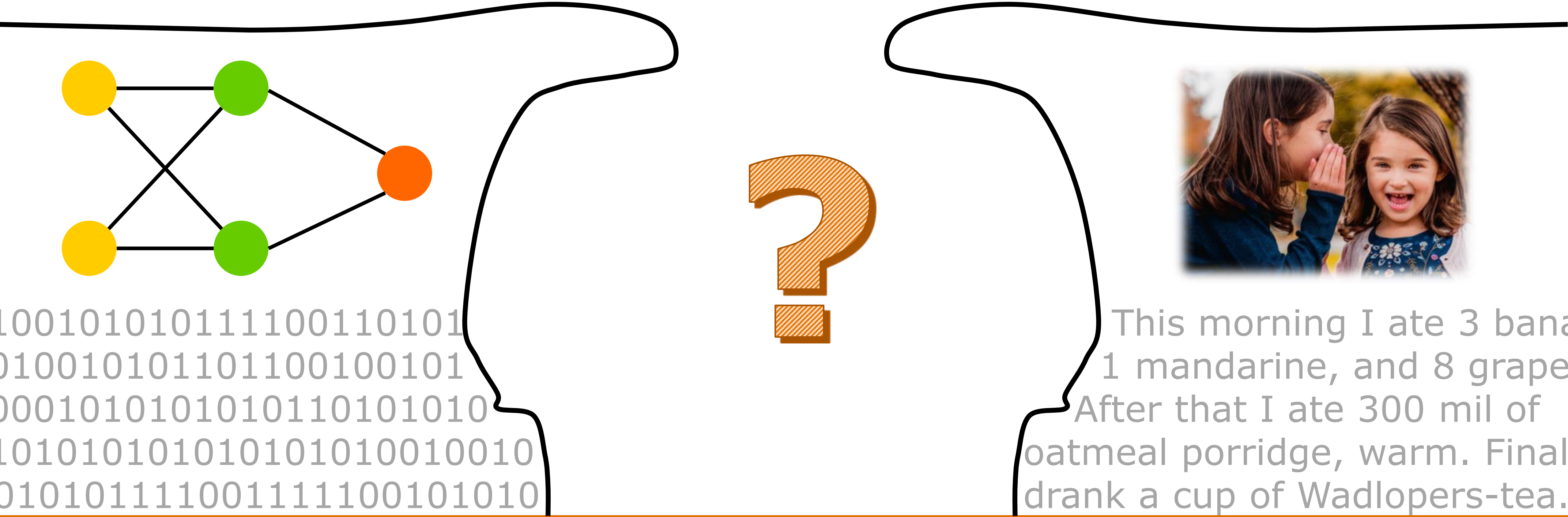
(C1) How does an ANN work with words?



Natural language processing: Chinese Whisper sentence



How does an ANN work with words?



Plenary session 4 : Turning a sentence into building blocks

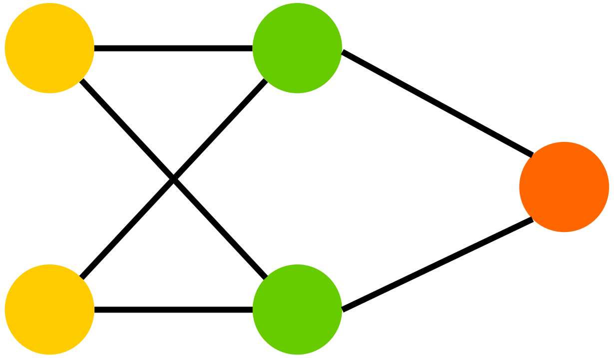
This morning I ate 3 bananas, 1 mandarine, and 8 grapes. After that I ate 300 mil of oatmeal porridge, warm. Finally, I drank a cup of Wadlopers-tea. What did you have for breakfast?

Word token

Attention head

Some are linked... 3 – bananas; 1 - mandarine

How does an ANN work with words?



100101010111100110101
0100101011011001001011
0001010101010110101010
101010101010101010010010
010101111001111100101010

This morning I ate 3 bana
1 mandarine, and 8 grape
After that I ate 300 mil of
oatmeal porridge, warm. Final
drank a cup of Wadlopers-tea.

Probability of tokens

The LLM needs to grasp the meaning of the token "WORK"

It observes the token "WORK" in its context using enormous amounts of training data

The nearby tokens are relevant while training

Source: https://ig.ft.com/generative-ai/

From token to vector

Upon first training we get a large set of tokens

- That are found adjacent to "WORK"

As well as tokens

- That were NOT found adjacent to "WORK"

The model then processes these tokens, not as letters, but as a **vector** (a list of values)

The more often a token is adjacent to WORK score higher, the ones not found adjacent score low – *probability score*

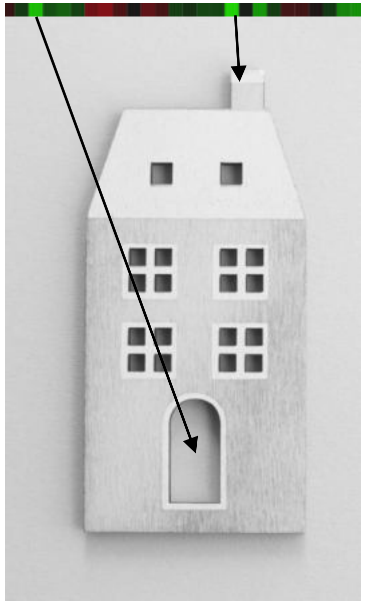
Vectors are a long sequence of values

A **vector** within a LLM can have many values

.. describing all the characteristics / **features of a token**

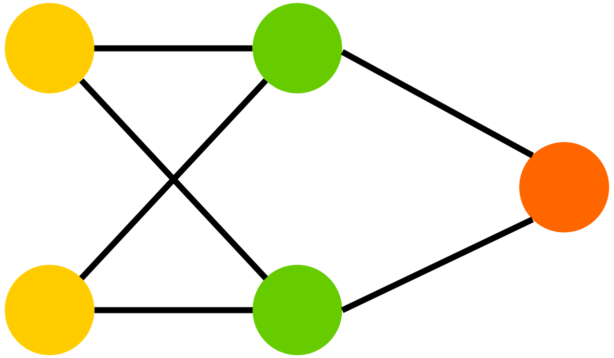
.. like a house:

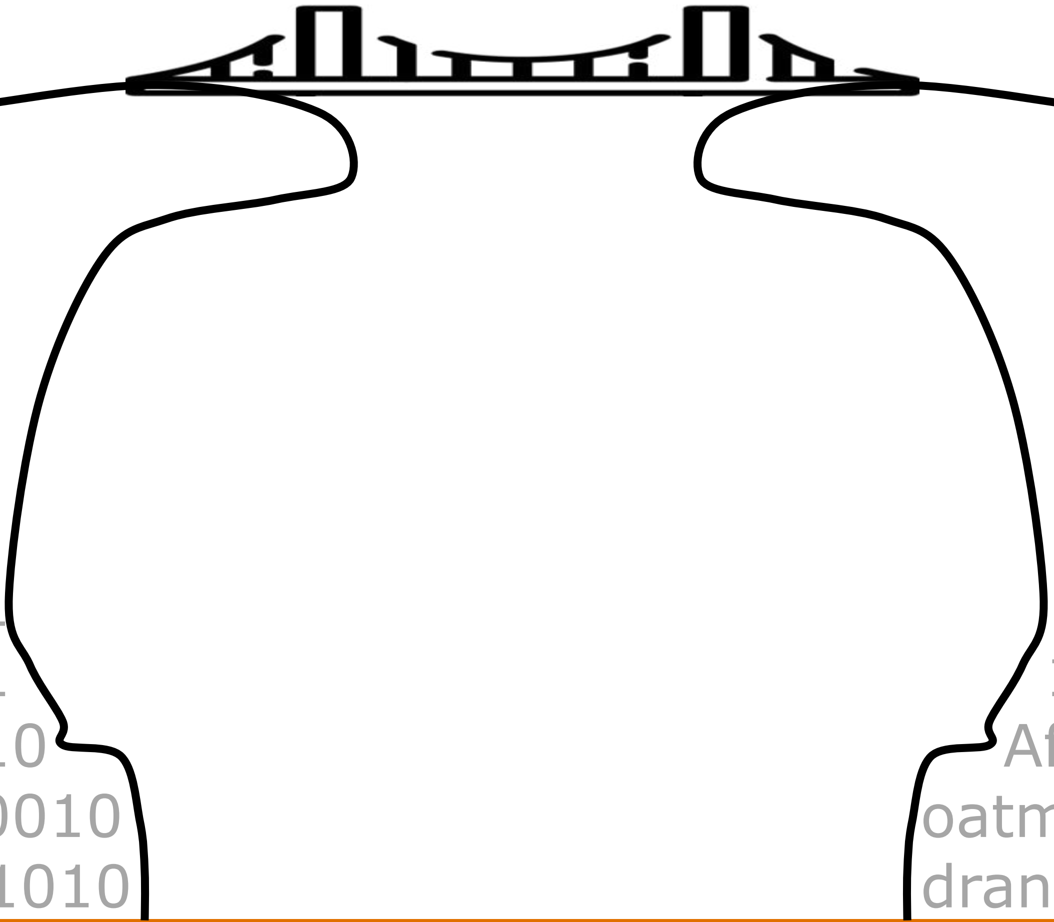
- Number of windows, doors, rooms
- Materials of the roof, walls
- Sizes, angles, positions
- Types of rooms



All linguistic features are turned into **values**

Converting words to values (vectors) bridges the gap





"Attention is all you need"

- June 2017 paper from Google Brain
- Available at arxiv.org

Crucial breakthrough for current LLMs:

- Transformer: new network architecture, based on:
  - Attention heads; and
  - Allowing parallelizable training
- Resulting in *outperforming all previous language models*

Attention Is All You Need

Ashish Vaswani*
Google Brain
avaswani@google.com

Noam Shazeer*
Google Brain
noam@google.com

Niki Parmar*
Google Research
nikip@google.com

Jakob Uszkoreit*
Google Research
usz@google.com

Llion Jones*
Google Research
llion@google.com

Aidan N. Gomez* †
University of Toronto
aidan@cs.toronto.edu

Łukasz Kaiser*
Google Brain
lukaszkaiser@google.com

Illia Polosukhin* ‡
illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

Recap training natural language

Transformer

Text input — Limitation: size context window

Tokenisation — Tokens are assigned

Index-based encoding — Each token gets its own index

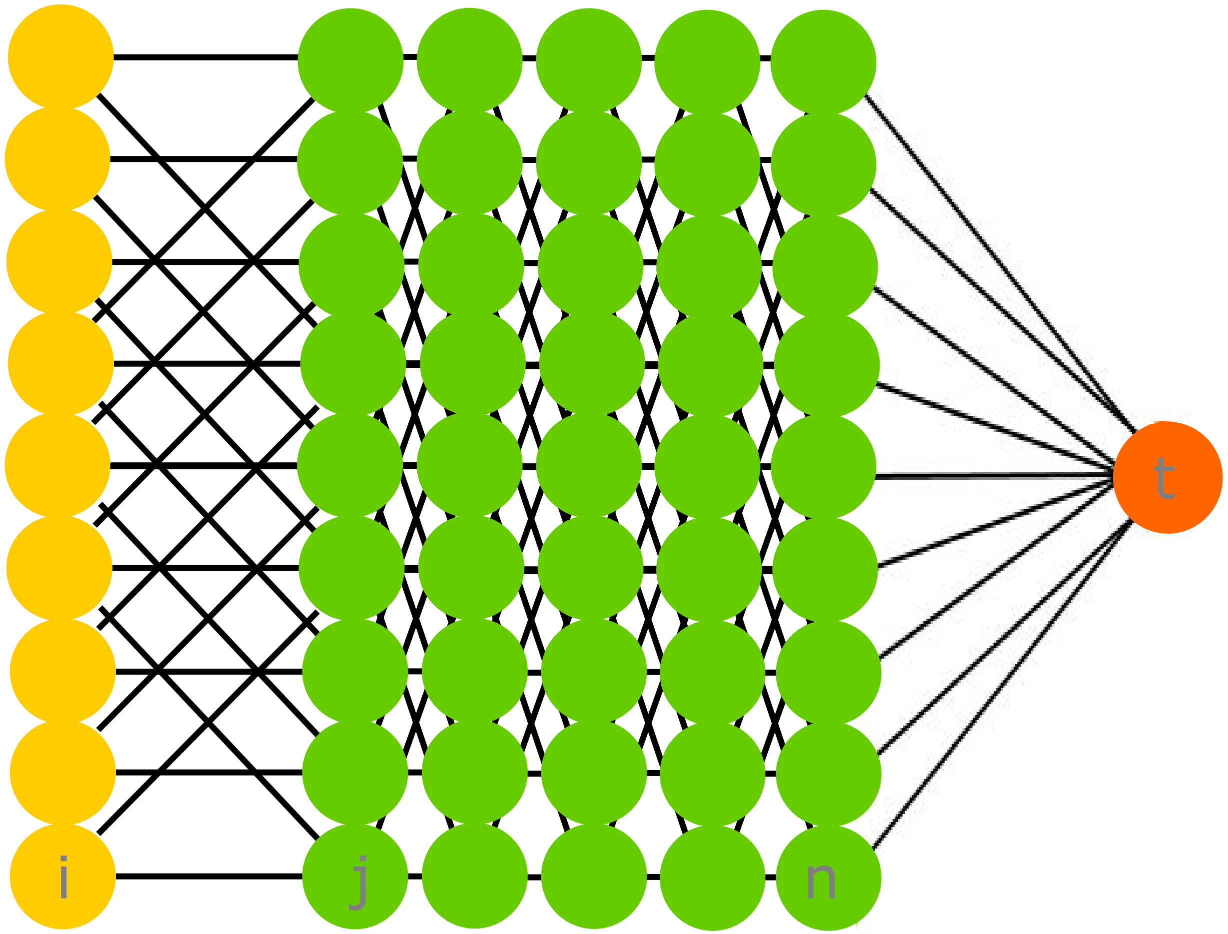
Embedding — Each token gets a vector

ENcoding

Positional encoding — Info on position in sentence, *added* to vector

Contextual encoding — Info on context — Attention heads

ANN for LLM (NLP)



Flexible number of inputs, **tens to hundreds HL**, ***variable*** output

Input is largely **<u>un</u>**structured

Pattern complexity: very complex

175 billion parameters to be finetuned (GPT-3)

Aspect	***Large Language Model (Transformer)***	***Language Model***
Dependency on distance in text	Can model long-distance dependencies in text thanks to their attention heads.	Often struggle with modeling long-distance dependencies.
Positional encoding	Use positional encoding to maintain the order of words in a sentence.	May depend on the order of the input, but often have no explicit positional encoding.
Notion of order	Maintain the notion of order through positional encoding.	The notion of order can be lost, depending on the architecture.

(C2) Latest LLM Developments

Retrieval augmented generation

- Up-to-date and domain-specific information is being incorporated. sources can be accessed via hyperlinks.
- The LLM can perform self-reflection by comparing its outputs with external information.

Ensembles

- Various, alternative models (LLM + other) are combined

✓ As a result, better predictive performance can be obtained
✓ Experts in the field of AI believe that the real power of transformers and attention heads lies *beyond* language

Time will tell... the role of information specialists

As time progresses, the number of new publications which crowd every technology space will make traditional searching a more difficult task. This is a matter we should all take very seriously.

Together, we need to continuously evaluate the most reliable and cost-efficient tools and methods to deal with this ever-growing body of searchable literature.

Main take-aways messages (part C)

- Neural networks can handle texts as the texts are converted into values, by first splitting up texts into word tokens, then assigning vectors.
- Vectors, within LLMs, contain the properties, probabilities of the token.
- Vectors also contain information on the position of the token in a sentence.
- Context matters within LLMs. So do attention heads and transformers:
  This is possible due to the developments at Google Brain (2017).
- The resulting ANNs contain many HLs and parameters.

- In the future LLMs are expected to improve even further.

(C3) Hypothesis 1

"For an information specialist basic insights into the working of LLMs is indispensable"

Hypothesis 2

"Information specialists have the task to promote LLM literacy within their organisation"

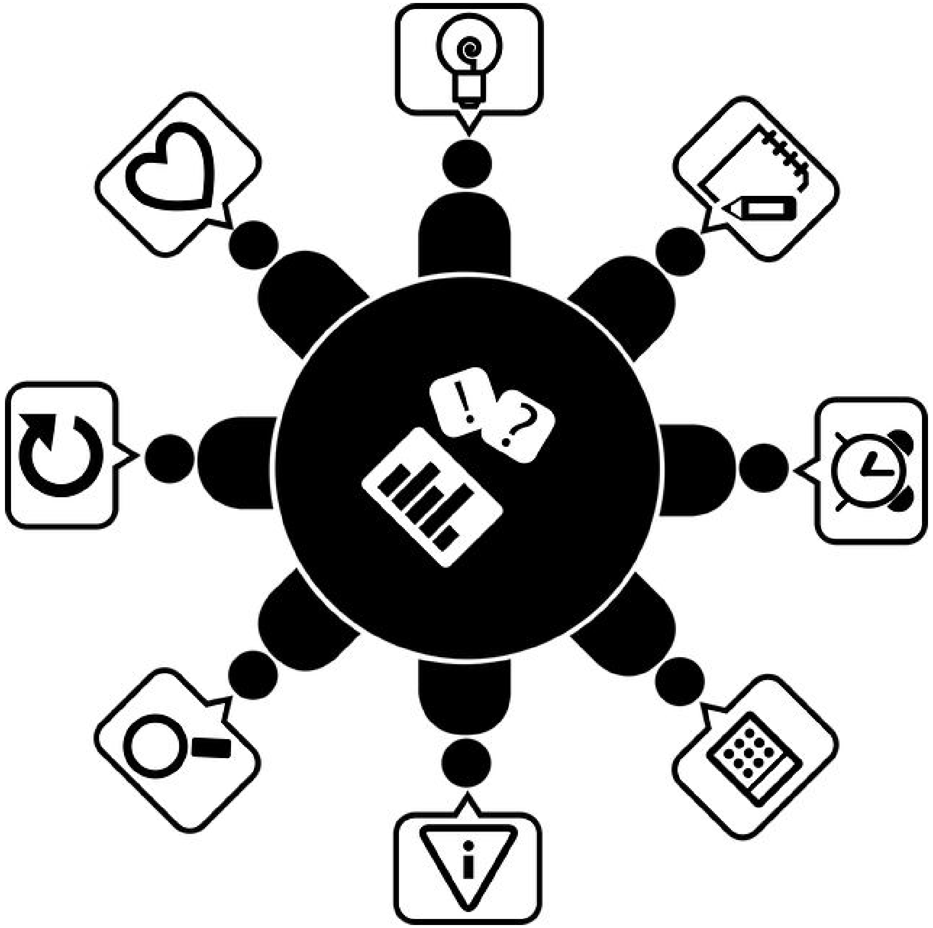
It is time for Questions!

(C4) Plenary session 5

What is (y)our role on “what happens under the hood” of LLMs / genAI?

What do we aim to teach others?

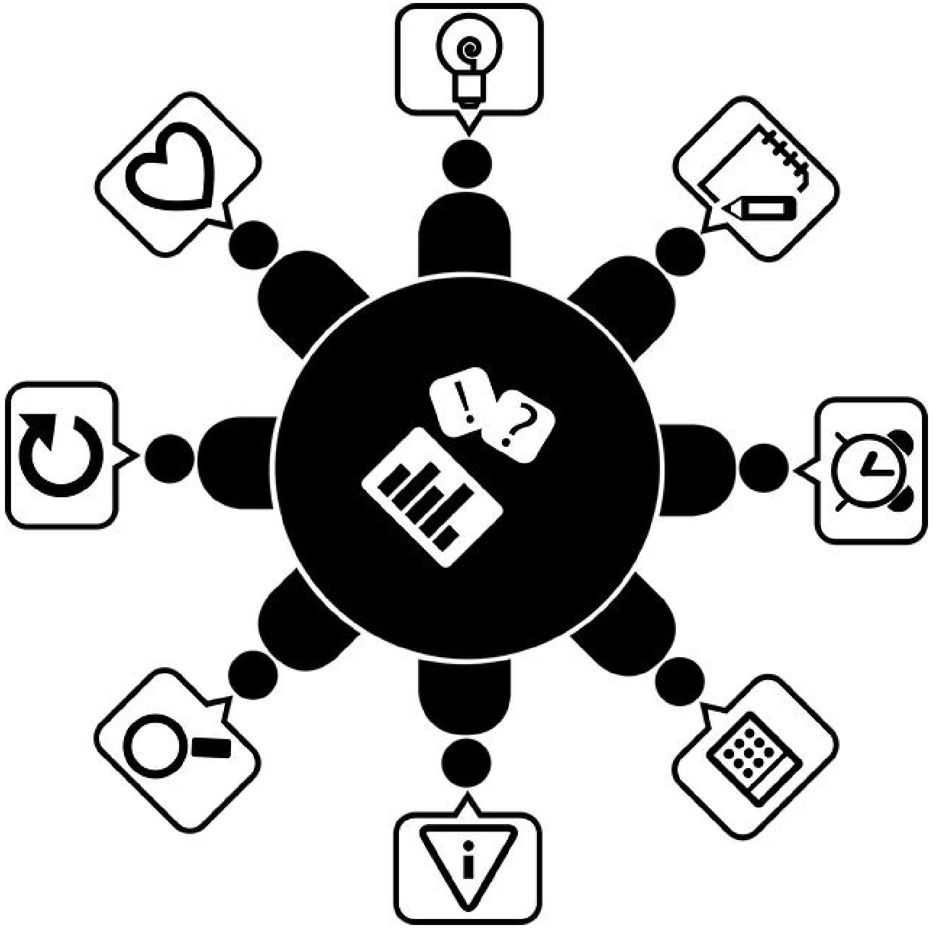
How / where?



Plenary session 6

What analogies do you see between the jargon of information professionals and the jargon of LLMs / genAI?

How / where?



Source: Pixabay

Wat weet jij over wat er bij een groot taalmodel onder de motorkap gebeurt?


TIME TO SAY GOODBYE
Contact: HC.Krijnsen@mindef.nl

(D) Further reading

AI voor docenten

LLM basics
(studenten)

Prompt eng gids

KULeuven
AI rules

European
AI Act

The Google paper

Brilliant video
On LLMs

Prompt eng
LLM Course

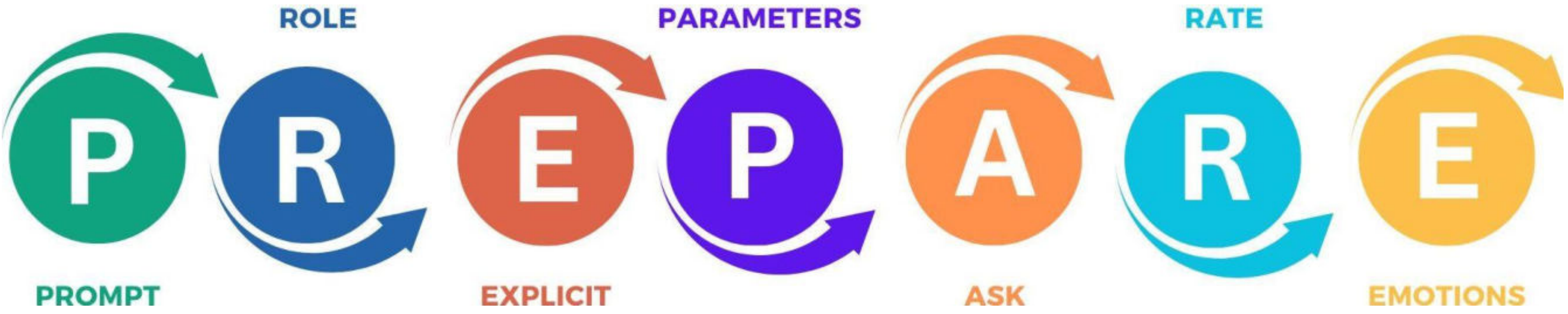
Reasoning engine versus search engine

The reasoning engine requires so-called 'prompt engineering' skills:

You need to provide clear, detailed instructions and hone your prompt

- be specific
- provide context (incl. examples)
- break things down
- use clear language
- experiment (iterate)
- know the lingo (domain specific)

Prompting is crucial



Strong prompts lead to more relevant answers

Remain critical: to answers, to generated images, to the output style...

Source: The AI Classroom, D. Fitzpatrick

Sources

Various sources have been used to come to this presentation: YouTube, LinkedIn, Coursera and Microsoft courses, and news flashes.

Further, some parts of the Ph.D. thesis “Advanced control of NOx diesel emissions” by myself, Henrike Krijnsen, 2000 were used to more clearly explain parts of section (B).

The colours as used in the ‘Neural Network Zoo’ (https://www.asimovinstitute.org/neural-network-zoo/) were used to illustrate the neural network examples as given in this presentation.

Images

All images in this presentation are either

- Made by the presenter, H.C. Krijnsen,
- Generated by AI, making use of Dall-E 3, or
- From Pixabay - used with permission
- Taken from the MS picture database
- Icons from https://www.iconfinder.com/ – no attribution required

Video

Source: fragment taken from https://ig.ft.com/generative-ai/

The audio of the ‘self’-test video originates from Pixabay