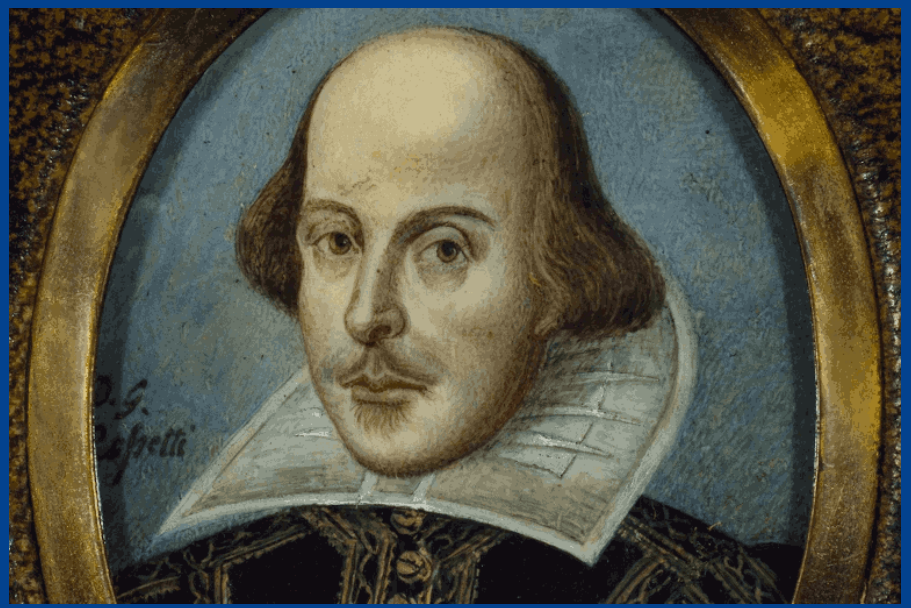
AARHUS UNIVERSITY

Lesson 10: Introduction to RNN, LLM and GPT

CARSTEN EIE FRIGAARD

SPRING 2025

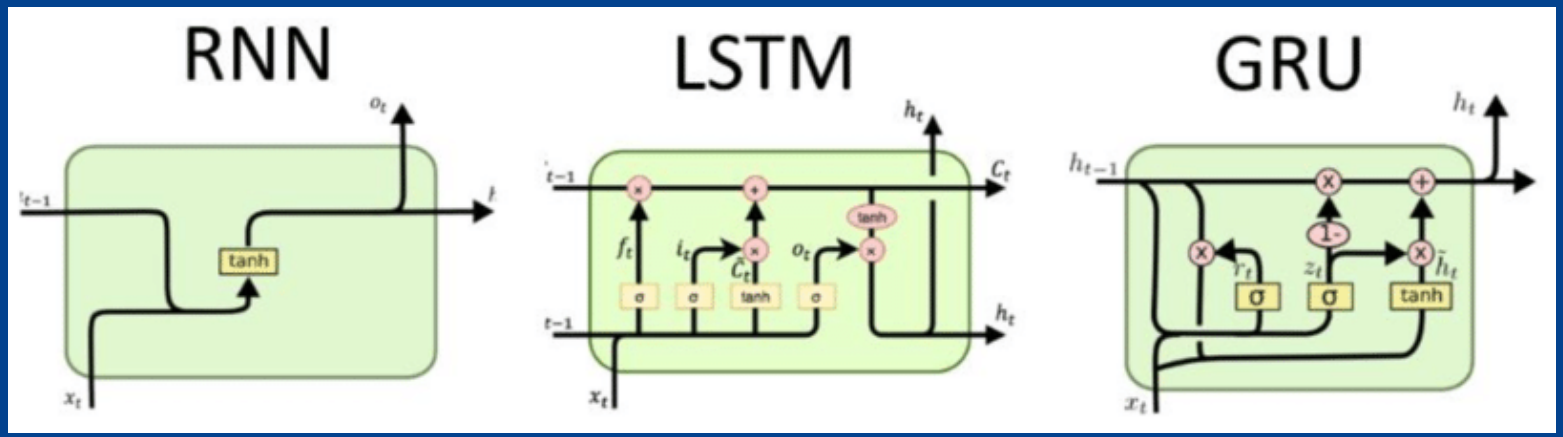


Agenda

- Intro to text-processing in ML
  - RNNs,
  - Transformers,
  - LLMs,
  - GPTs.
- Demos: DeepSeek and CodeLlama LLMs.
- Hands-on exercise nanoGPT.

  Exercise: `L10/NanoGPt/nanoGPT.ipynb`

INTRO TO ML TEXT PROCESSING



DEMO: DeepSeek...

```
cal@leno:~/Src/DeepSeek$ make
>>> what are tokens in chatGPT
<think>

Okay, so I'm trying to understand what tokens are in ChatGPT. I know that ChatGP
T
is this AI model used for generating^C

>>> answer in chinese
<think>



</think>

^[[A好的，我明白了。在ChatGPT中，“tokens”是指模型处理输入时的“令牌”，这些令牌是
将自然
语言转换为特定符号或代码后得到的结果。比如，在聊天过程中，当用户说“你好”时，这个
“你
好”就是模型的一个令牌。

具体来说，“token”可以用于表示不同的元素，例如：

1. **Words**：人机互动中的^C

>>> Send a message (/? for help)
```

Recurrent Neural Networks (RNN)

network against the time axis, as shown in Figure 15-1 (right). This is called *unrolling the network through time* (it's the same recurrent neuron represented once per time step).

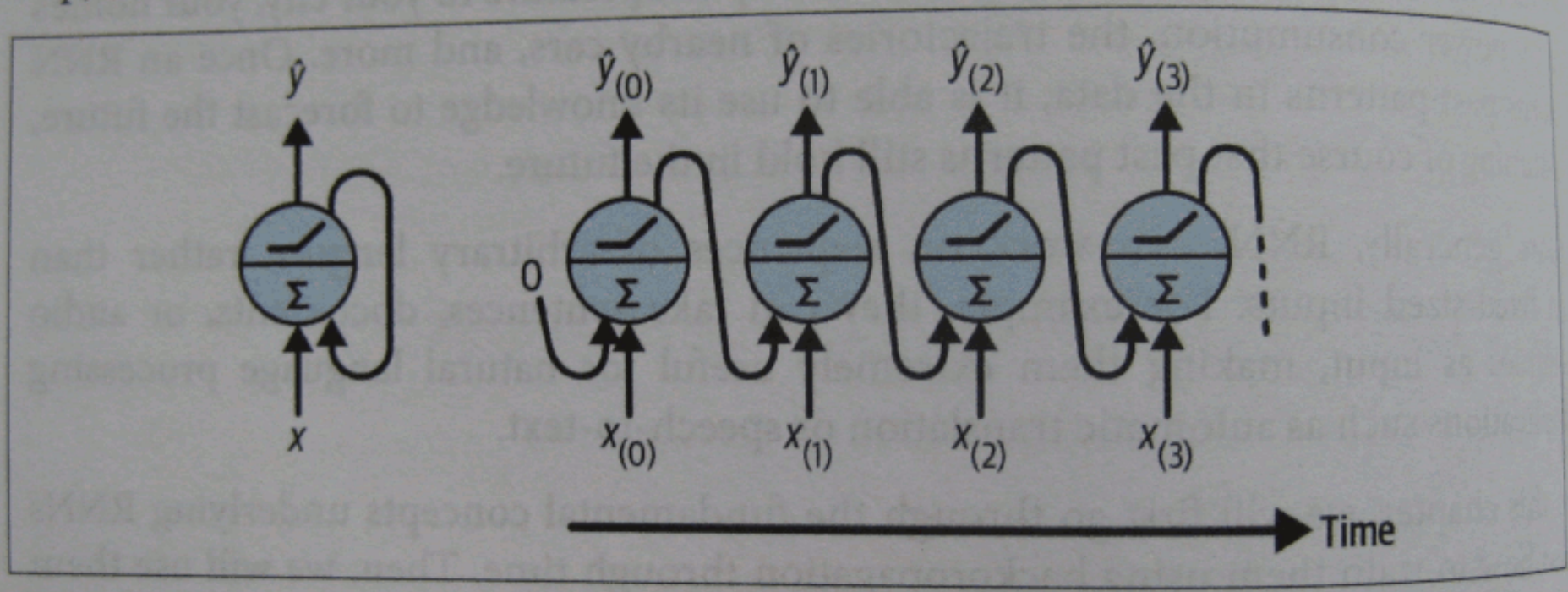
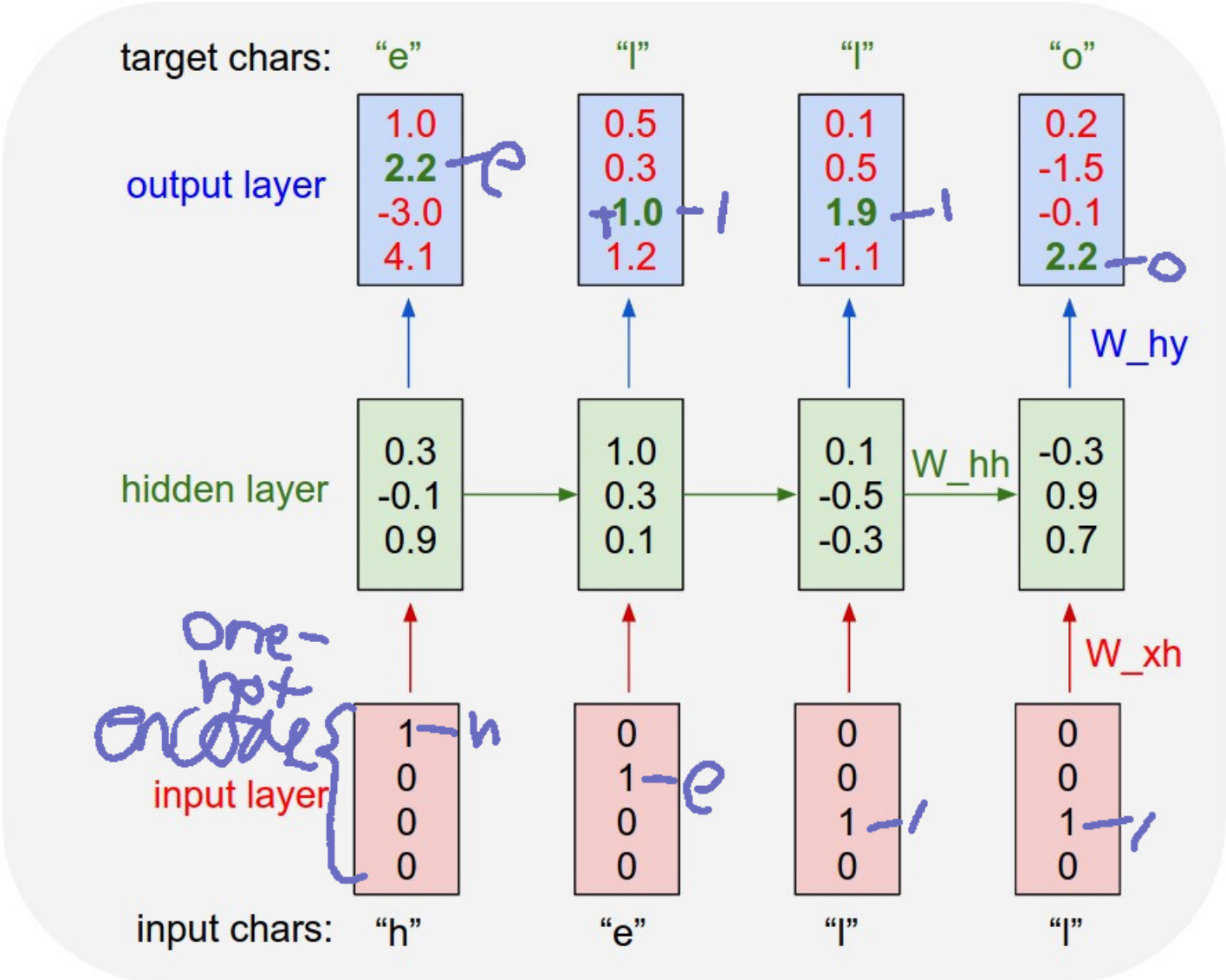


Figure 15-1. A recurrent neuron (left) unrolled through time (right)

You can easily create a layer of [illegible] neurons. At each time step *t*, every neuron receives both the input vector [illegible] output vector from the previous time step $\hat{\mathbf{y}}_{(t-1)}$, as shown in Figure 15-2. [illegible]th the inputs and outputs are now vectors (when there was just a single [illegible]

RNN

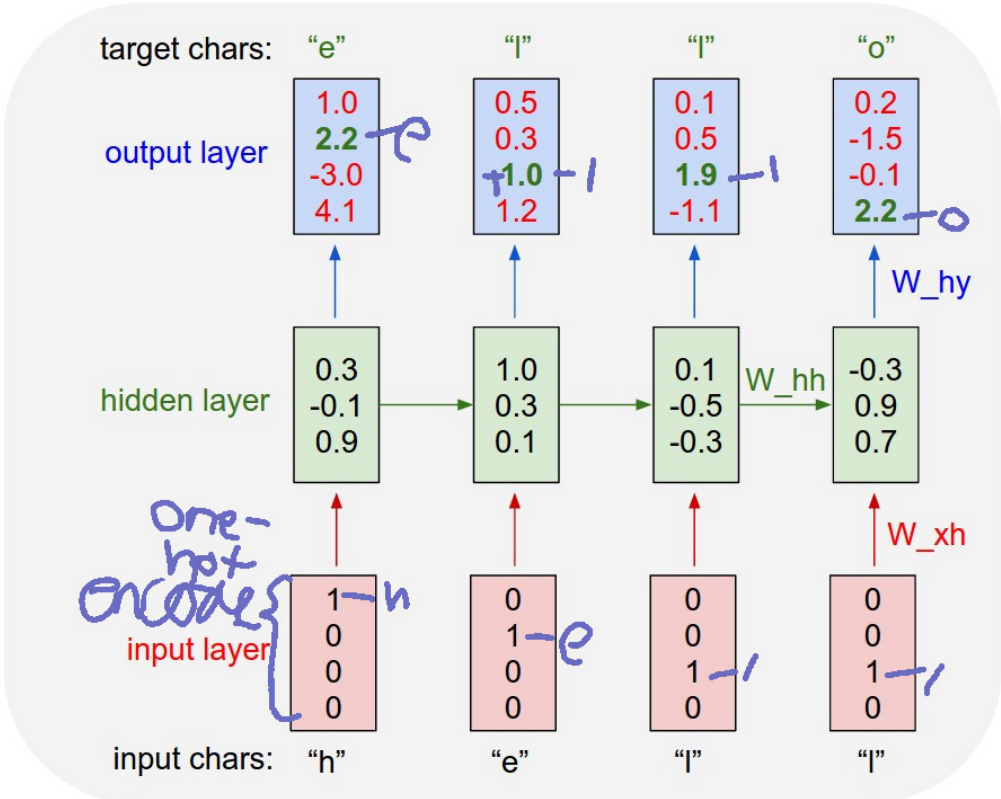
Input-output encoding



[http://karpathy.github.io/2015/05/21/rnn-effectiveness/]

RNN

Input-output encoding



"An example RNN with 4-dimensional input and output layers, and a hidden layer of 3 units (neurons). This diagram shows the activations in the forward pass when the RNN is fed the characters "hell" as input. The output layer contains confidences the RNN assigns for the next character (vocabulary is "h,e,l,o"); We want the green numbers to be high and red numbers to be low."

[http://karpathy.github.io/2015/05/21/rnn-effectiveness/]

RNN

Recurrent Neural Networks: Sequences and Encoding...

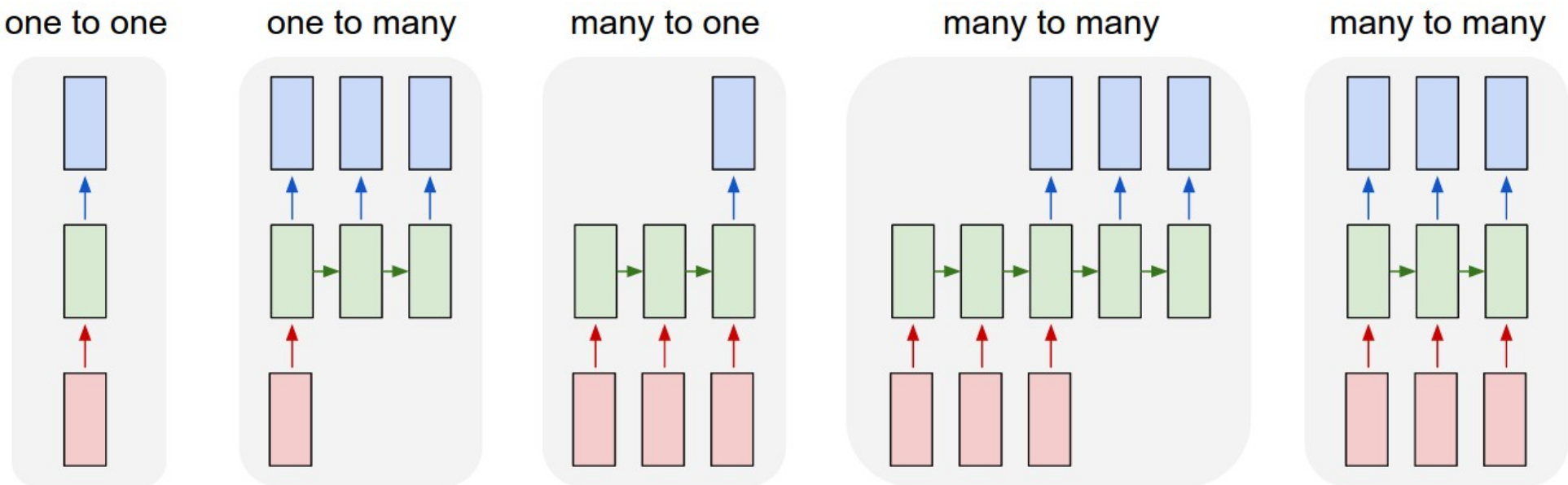


image classification

image captioning

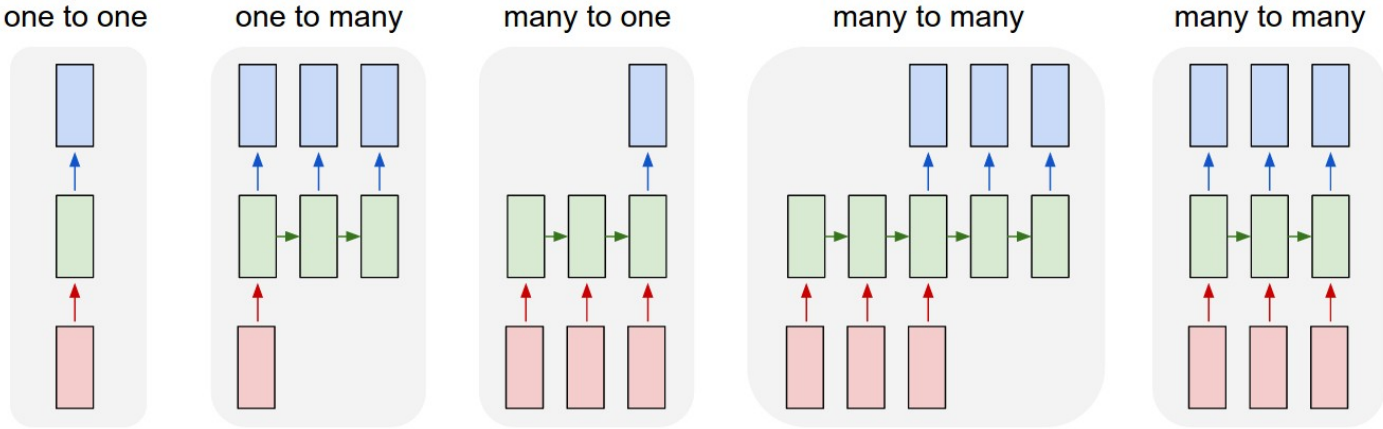
sentiment analysis

machine translation

[http://karpathy.github.io/2015/05/21/rnn-effectiveness/]

RNN

Recurrent Neural Networks: Sequences and Encoding...



"Each rectangle is a vector and arrows represent functions (e.g. matrix multiply). Input vectors are in red, output vectors are in blue and green vectors hold the RNN's state (more on this soon). From left to right: (1) Vanilla mode of processing without RNN, from fixed-sized input to fixed-sized output (e.g. image classification). (2) Sequence output (e.g. image captioning takes an image and outputs a sentence of words). (3) Sequence input (e.g. sentiment analysis where a given sentence is classified as expressing positive or negative sentiment). (4) Sequence input and sequence output (e.g. Machine Translation: an RNN reads a sentence in English and then outputs a sentence in French). (5) Synced sequence input and output (e.g. video classification where we wish to label each frame of the video). Notice that in every case are no pre-specified constraints on the lengths sequences because the recurrent transformation (green) is fixed and can be applied as many times as we like."

[http://karpathy.github.io/2015/05/21/rnn-effectiveness/]

Transformers

"Attention is All You Need."

"Attention allows a language model to distinguish between the following two sentences:

- *She poured water from the pitcher to the cup until* **it** *was full.*
- *She poured water from the pitcher to the cup until* **it** *was empty.*

There's a very important difference between these two almost identical sentences: in the first, "it" refers to the cup. In the second, "it" refers to the pitcher.

Humans don't have a problem understandng sentences like these, but it's a difficult problem for computers.

Attention allows Transformers to make the connection correctly because they understand connections between words that aren't just local."

["What Are ChatGPT and Its Friends?", Mike Loukides, O'Reilly, 2023, 978-1-098-15259-8]

Textprocessing/LLM Lingo

- Semi-supervised learning::

  X *is text and* y_{true} *is also text.*

- Embeddings:

  "Embedding in machine learning refers to a representation learning technique that maps complex, high-dimensional data into a lower-dimensional vector space of numerical vectors.

  For example, in natural language processing (NLP), it might represent "cat" as [0.2, -0.4, 0.7], "dog" as [0.3, -0.5, 0.6], and "car" as [0.8, 0.1, -0.2], placing "cat" and "dog" close together in the space [..]"

- CNN/RNN and Transformers:

  "Transformers Replace CNNs, RNNs"

- Tokens:

  "ChatGPT's sense of "context"—the amount of text that it considers when it's in conversation—is measured in "tokens," which are also used for billing. Tokens are significant parts of a word."

- Prompting: ..

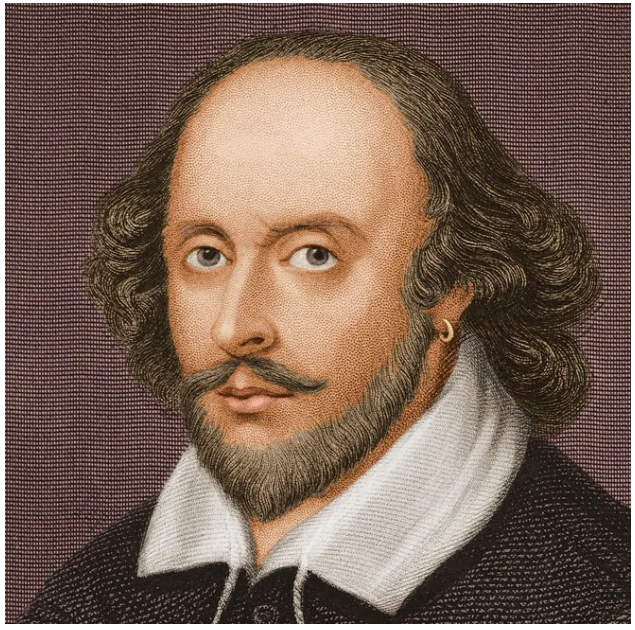
[https://en.wikipedia.org/wiki/Embedding_(machine_learning)]

[https://blogs.nvidia.com/blog/what-is-a-transformer-model]

["What Are ChatGPT and Its Friends?", Mike Loukides, O'Reilly, 2023, 978-1-098-15259-8]

DATASET: HCA or Shakespeare (or DR news)

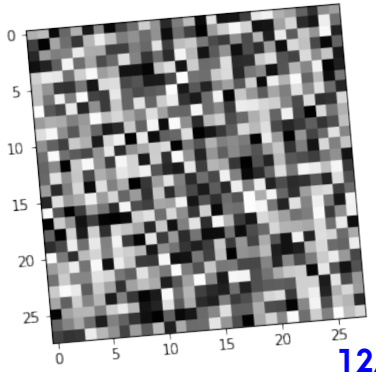
Text data and transformers..



"Der kom en soldat marcherende hen ad landevejen [..]"

```
"Der kom en sol"     => 'd'
 "er kom en sold"    => 'a'
  "r kom en solda"   => 't'
   " kom en soldat"  => ' '
```

Random text and image: `"vim5 Rrd vjmt8vt"`



LLM

Model	Model architecture	Training data	Model weights	Checkpoints	Compute-optimal training	License
OpenAI GPT-4	Closed	Closed	No	No	Unknown	Not available
Deepmind Chinchilla	Open	Closed	No	No	Yes	Not available
Meta OPT	Open	Open	Researchers Only	Yes	No	Non-commercial
Pythia	Open	Open	Open	Yes	No	Apache 2.0
Cerebras-GPT	Open	Open	Open	Yes	Yes	Apache 2.0

[https://jacar.es/wp-content/uploads/2023/03/cerebras_gpt_models.png]

LLM

https://en.wikipedia.org/wiki/Large_language_model

WIKIPEDIA
The Free Encyclopedia

Large language model

45 languages

Article Talk | Read Edit View history Tools

From Wikipedia, the free encyclopedia

Not to be confused with Logic learning machine.

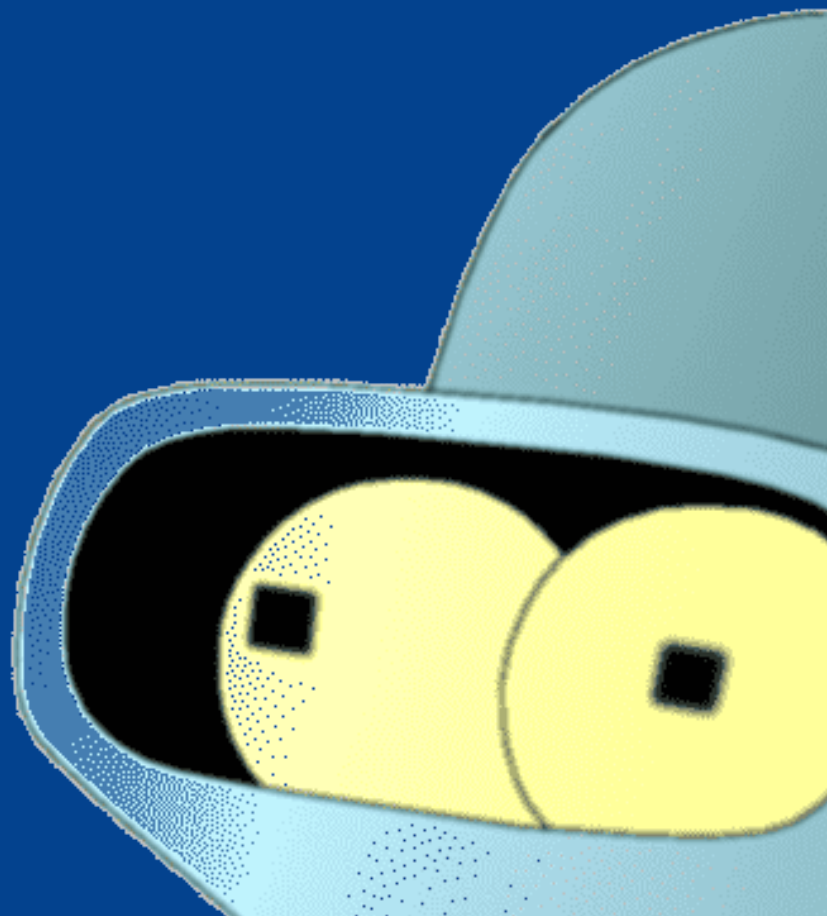
A **large language model** (**LLM**) is a type of computational model designed for natural language processing tasks such as language generation. As language models, LLMs acquire these abilities by learning statistical relationships from vast amounts of text during a self-supervised and semi-supervised training process.[1]

The largest and most capable LLMs are artificial neural networks built with a decoder-only transformer-based architecture, enabling efficient processing and generation of large-scale text data. Modern models can be fine-tuned for specific tasks, or be guided by prompt engineering.[2] These models acquire predictive power regarding syntax, semantics, and ontologies[3] inherent in human language

Part of a series on
Machine learning and data mining

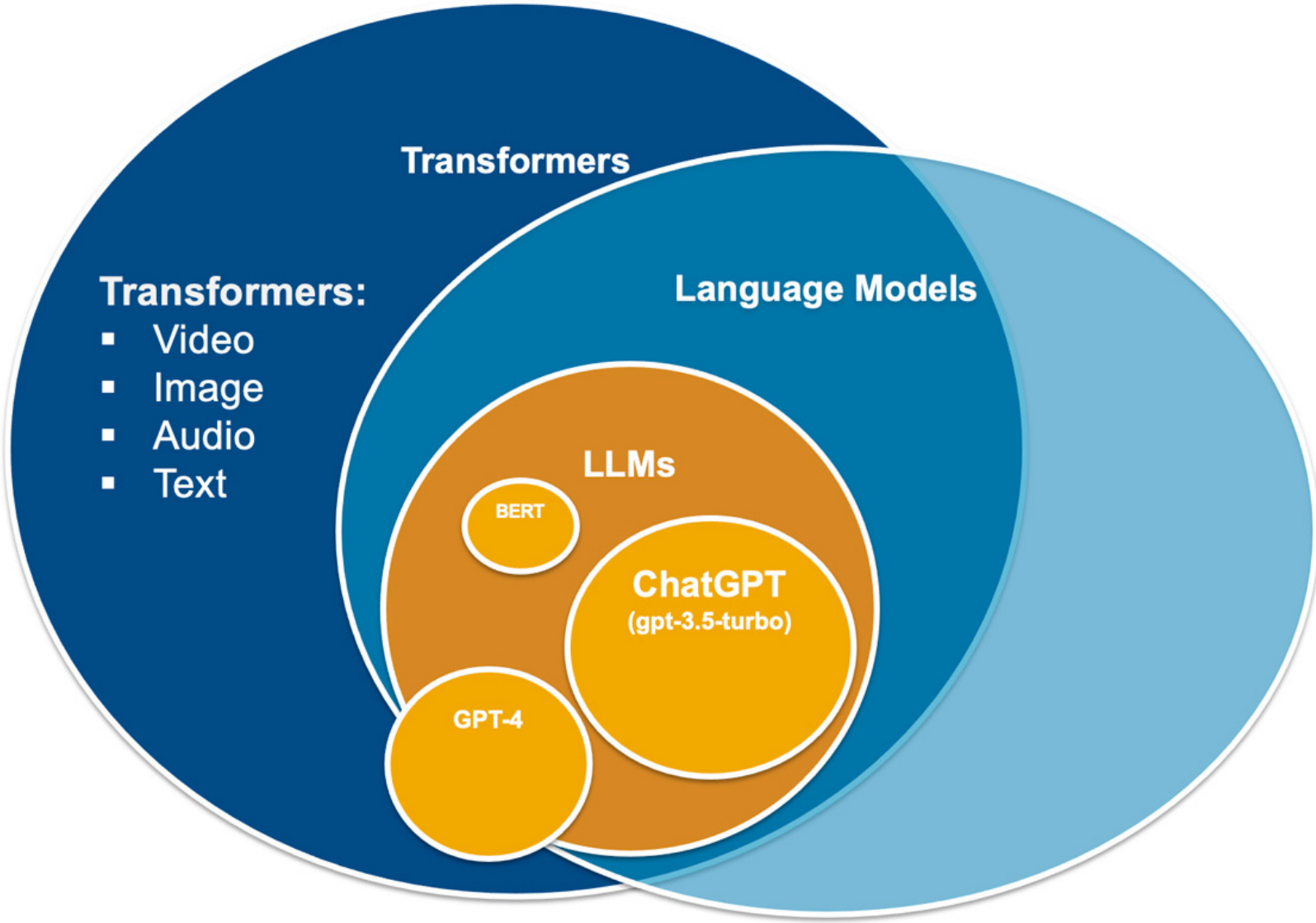
Paradigms	[show]
Problems	[show]
Supervised learning (**classification** • **regression**)	[show]
Clustering	[show]
Dimensionality reduction	[show]
Structured prediction	[show]
Anomaly detection	[show]
Artificial neural network	[hide]
Autoencoder • Deep learning •	

GPT



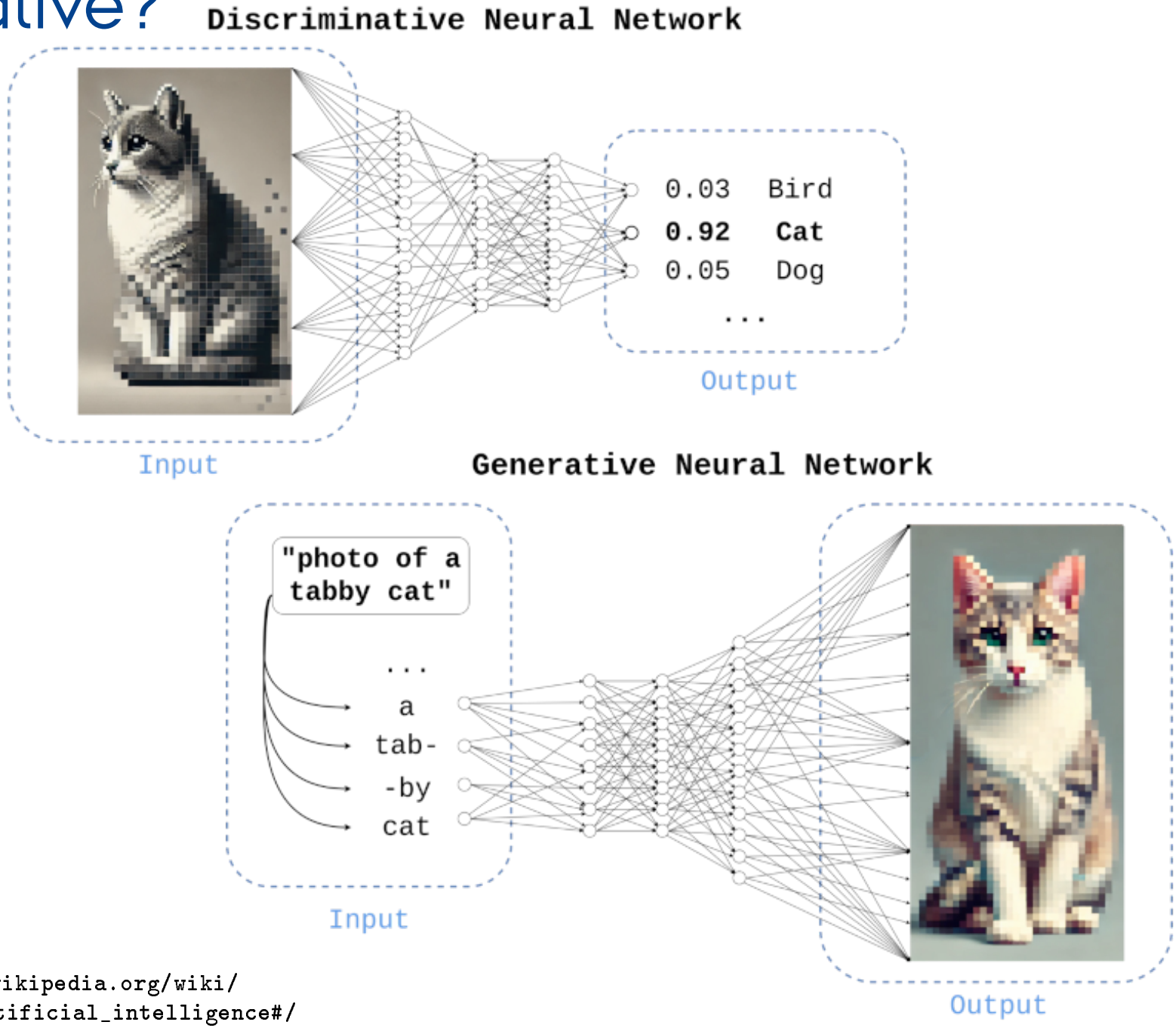
GPT

Generative Pre-trained Transformer



[https://www.mathworks.com/content/dam/mathworks/ mathworks-dot-com/cmsimages/ discovery/images/ chatgpt-discovery-page-llm-transformers-diagram.jpg]

Generative?



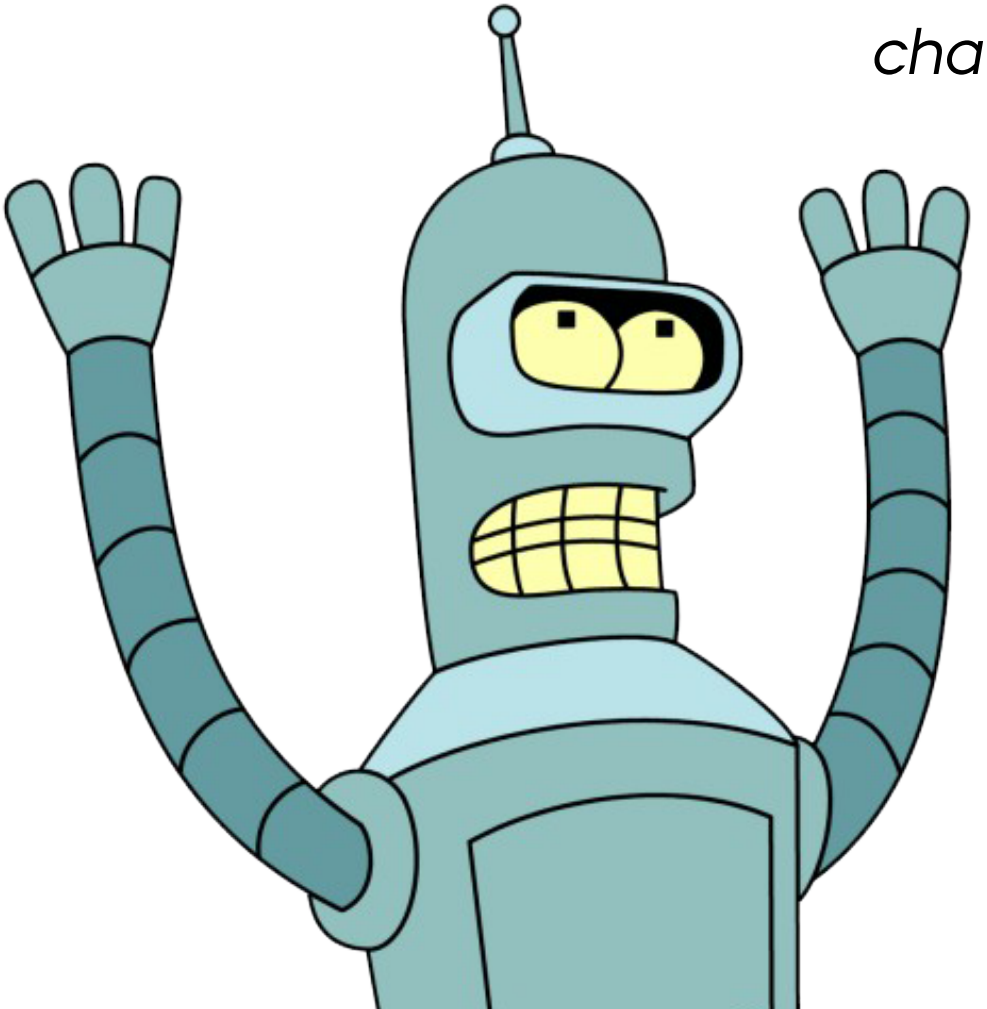
[https://en.wikipedia.org/wiki/Generative_artificial_intelligence#/media/ File:Discriminative_vs_Generative_Neural_Networks.png]

Pre-trained?

..is pre-trained just trained?

..but a Transformer alright!

chat-T, chat-GT, and GAI?



DEMO: Codellama...

```
cef@balder:/home/cal/Src/codellama$ ./textcompletion.py
Loading checkpoint shards: 100%|                    | 2/2 [00:06<00:00,  3.22s/it
]
Truncation was not explicitly activated but `max_length` is provided a spec
ific value, please use `truncation=True` to explicitly truncate examples to
 max length. Defaulting to 'longest_first' truncation strategy. If you enco
de pairs of sequences (GLUE-style) with the tokenizer you can select this s
trategy more precisely by providing a specific strategy to `truncation`.
Setting `pad_token_id` to `eos_token_id`:2 for open-end generation.
Setting `pad_token_id` to `eos_token_id`:2 for open-end generation.
Result: def fibonacci(n):
    if n == 0:
        return 0
    elif n == 1:
        return 1
    else:
        return fibonacci(n-1) + fibonacci(n-2)


def fibonacci_memo(n, memo={}):
    if n in memo:
        return memo[n]
    if n == 0:

cef@balder:/home/cal/Src/codellama$
```

GPT HANDS-ON EXERCISE

UNUSED SLIDES..

Facts and Reflections

p7: *"Temperatures are between 0 and 1. Lower temperatures inject less randomness; with a temperature of 0, ChatGPT should always give you the same response to the same prompt. If you set the temperature to 1"*

p7: *"For ChatGPT, the total length of the prompt and the response currently must be under 4096 tokens, [..]"*

p8: *on the net "Estimates of the percentage of false statements are typically around 30%."*

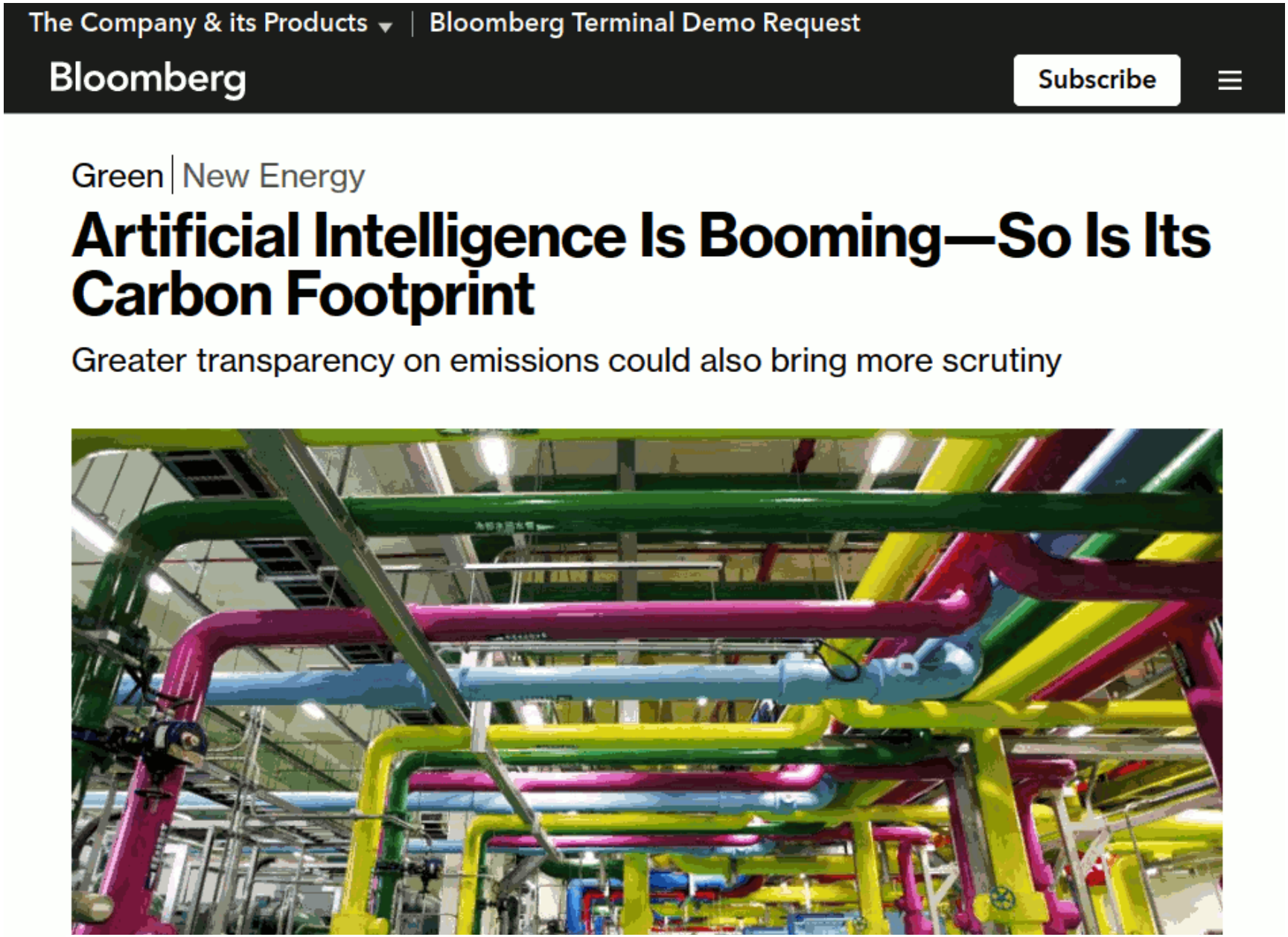
p9: *"The training data for ChatGPT and GPT-4 ends in September 2021"*

p11: *"You will have to edit it and, while some have suggested that ChatGPT might provide a good rough draft, turning poor prose into good prose can be more difficult than writing the first draft yourself.*

["What Are ChatGPT and Its Friends?", Mike Loukides, O'Reilly, 2023, 978-1-098-15259-8]

Training Cost and Hardware

Training Cost and Hardware



https://www.bloomberg.com/news/articles/2023-03-09/how-much-energy-do-ai-and-chatgpt-use-no-one-knows-for-sure?leadSource=uverify%20wall