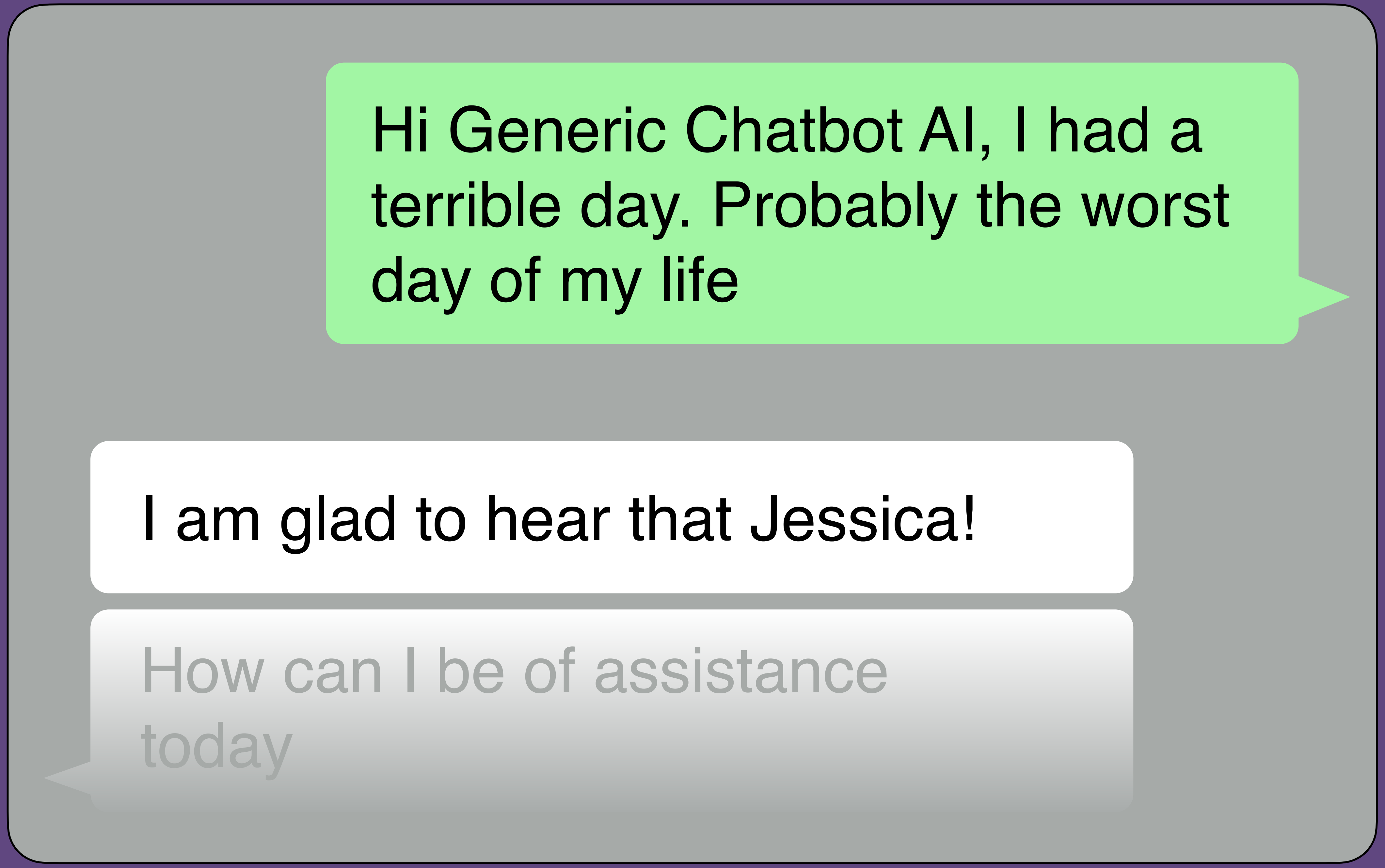
Transformer + a bit of NLP



Hi Generic Chatbot AI, I had a terrible day. Probably the worst day of my life

I am glad to hear that Jessica!

How can I be of assistance today

Practical Deep Learning for Science
13 June, 2023

מכון ויצמן למדע
WEIZMANN INSTITUTE OF SCIENCE

- Nilotpal

- My first plan was to
  - Code it live with chatGPT
  - *(Which what I kinda ended up doing anyway)*

transformer build transformer
I used the ~~stones~~ to ~~destroy~~ the ~~stones.~~

> *“A live demo is a disaster awaiting to happen.”*
>
> \- Socrates

“A live demo is a disaster awaiting to happen.”

- ~~Socrates~~

Sorry I lied :)

Tokenization + Embedding

Cause we cannot input words to the network directly

- Cannot use words/characters as input to the network
  - ➡ Need to tokenize

Tokenization

You → 0
Hello → 1
Book → 2
▪ ▪
▪ → ▪
▪ ▪

Map b/w vocabulary and tokens

How are you? → Encode → (15, 29, 0, 11)

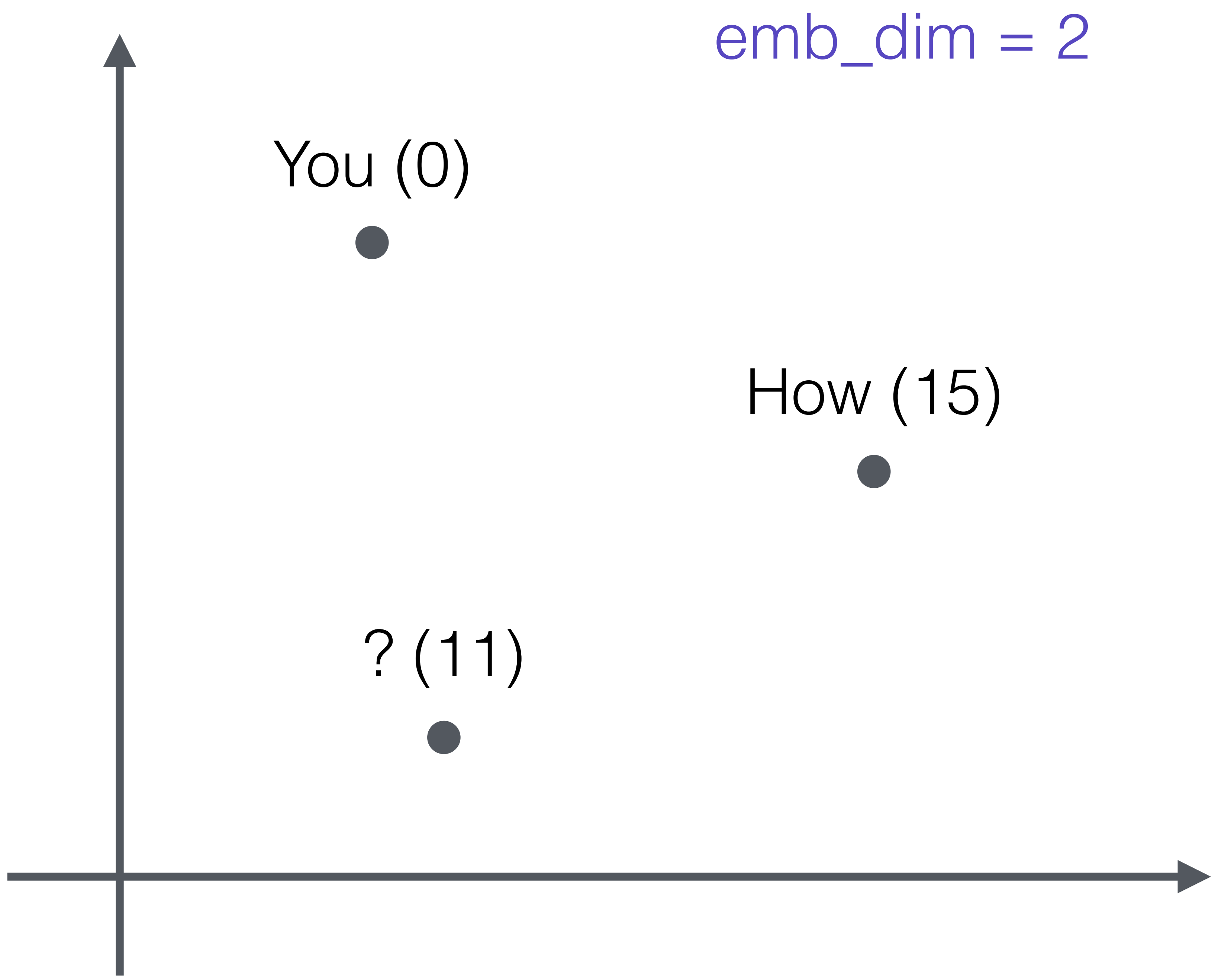
(15, 29, 0, 11) → Decode → How are you?

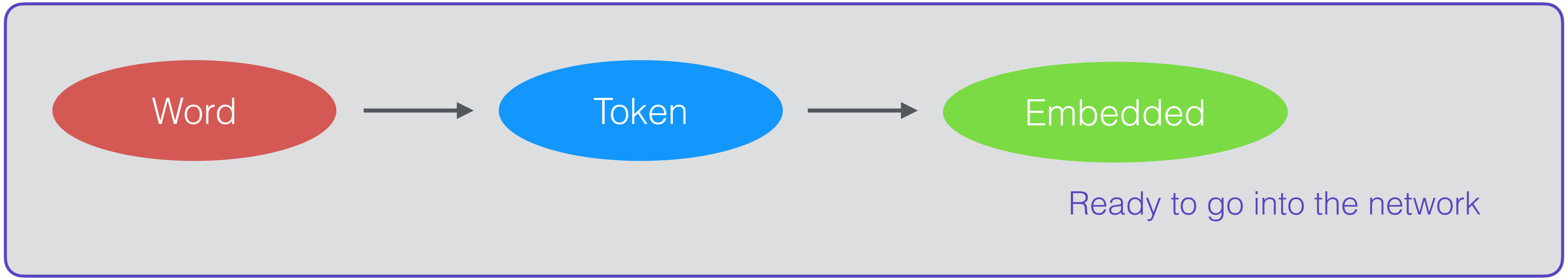
No Learning involved

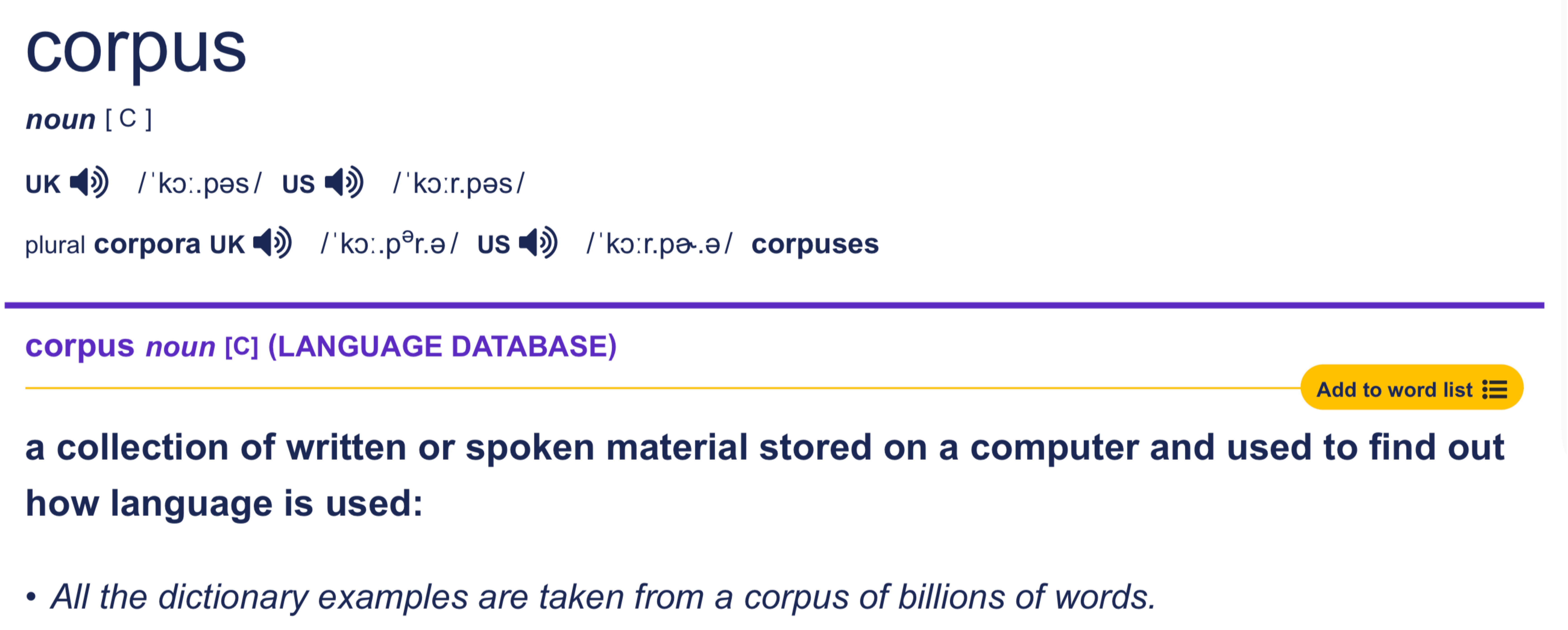
Lots of tokenizers available, but we will build our own

- The numbers don’t have an “ordering” meaning
  - ➡ You → 0, how → 15 *(quite random)*
  - ➡ *Need to embed them to some vector space*
  - ➡ **Learnable**
  - ➡ Matrix of shape (vocab_size, emb_dim) where each entry is a weight. (Can compute gradient)

`torch.nn.embedding(vocab_size, emb_dim)`







corpus

noun [C]

UK 🔊 /ˈkɔː.pəs/ **US** 🔊 /ˈkɔːr.pəs/

plural **corpora UK** 🔊 /ˈkɔː.pᵊr.ə/ **US** 🔊 /ˈkɔːr.pɚ.ə/ **corpuses**

---

corpus *noun* [C] (LANGUAGE DATABASE)

a collection of written or spoken material stored on a computer and used to find out how language is used:

- *All the dictionary examples are taken from a corpus of billions of words.*

Padding

- A sentence/querry is a sequence of tokens
  - Length can very
- Want to work with sequences of constant length
  - padding (usually last token + 1)
  - Attention mask
    - Vector that tells which tokens are real (1) which are pad (0)

How are you? → Tokenize → (15, 29, 0, 11) → Padding → (15, 29, 0, 11, 999, 999)

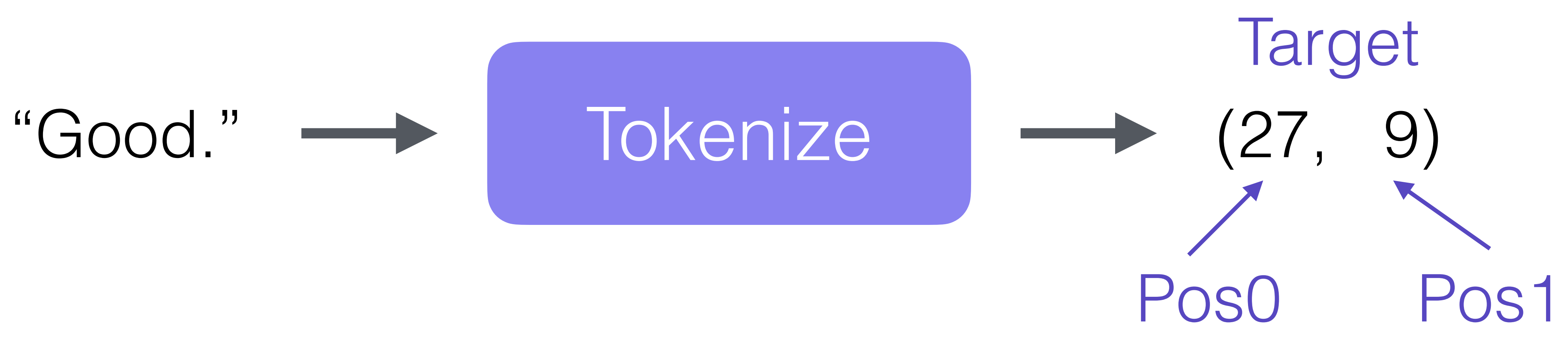
(1, 1, 1, 1, 0, 0)

Attention mask

Network predicts tokens

Think of it as a classification problem…

- Let’s say
  - Input: “How are you?”
  - Target: “Good.”
  - Vocabulary size = 1,000



“Good.” → Tokenize → (27, 9)

Target

Pos0 Pos1

- Each token is a class
- At each position, we’ll predict the class (token)
- Number of classes ~ total number of words! A lot!

<EOS> token

- Before padding we add a special token indicating the end of a response
  - ➡ Allows the network to predict responses of different lengths
  - ➡ Want to work with sequences of constant length, remember?

“<EOS>” → 998
“<PAD>” → 999

(Let’s say)

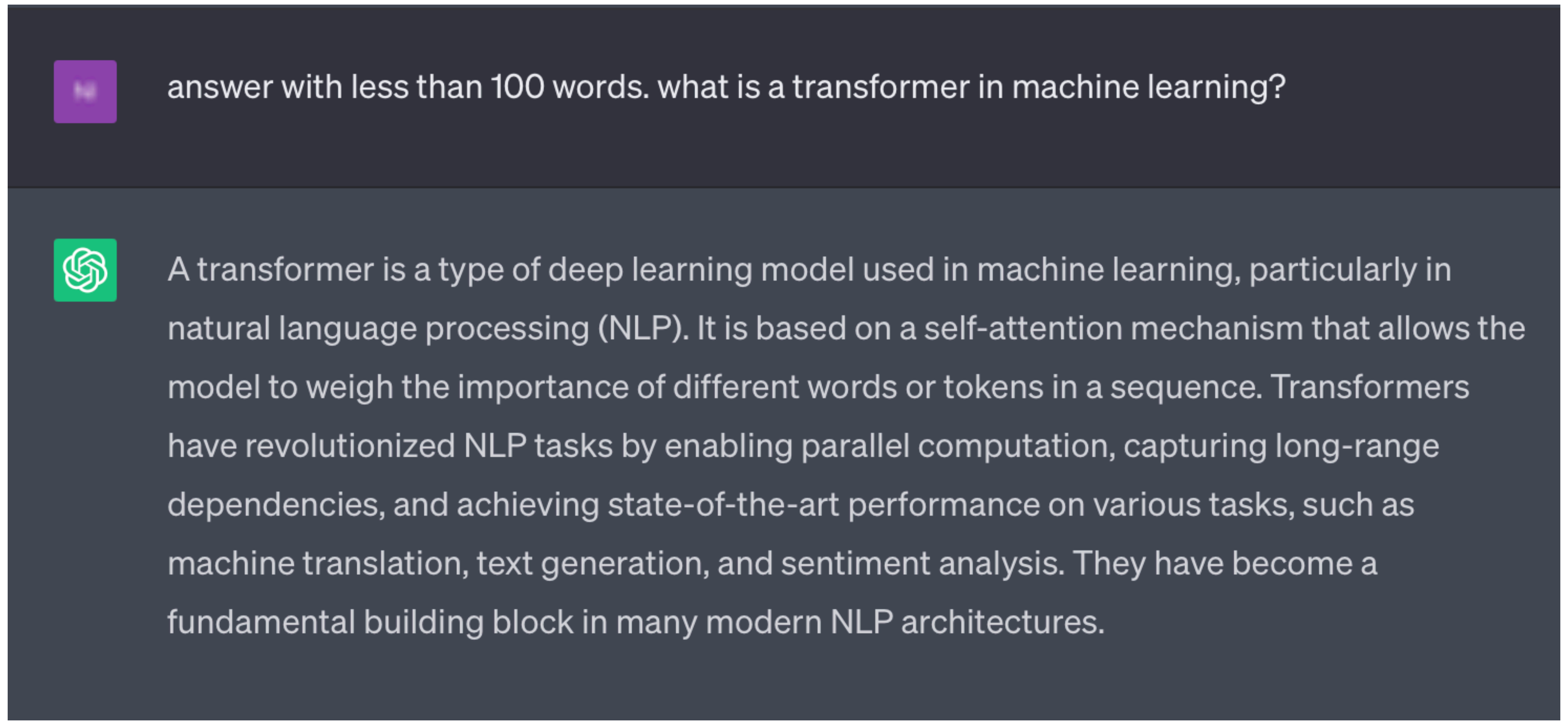
Let’s say we want to work with sequences of length 5

<EOS> <PAD>

“Good.” → Tokenize → (27, 9) → Padding → (27, 9, **998**, **999**, **999**)

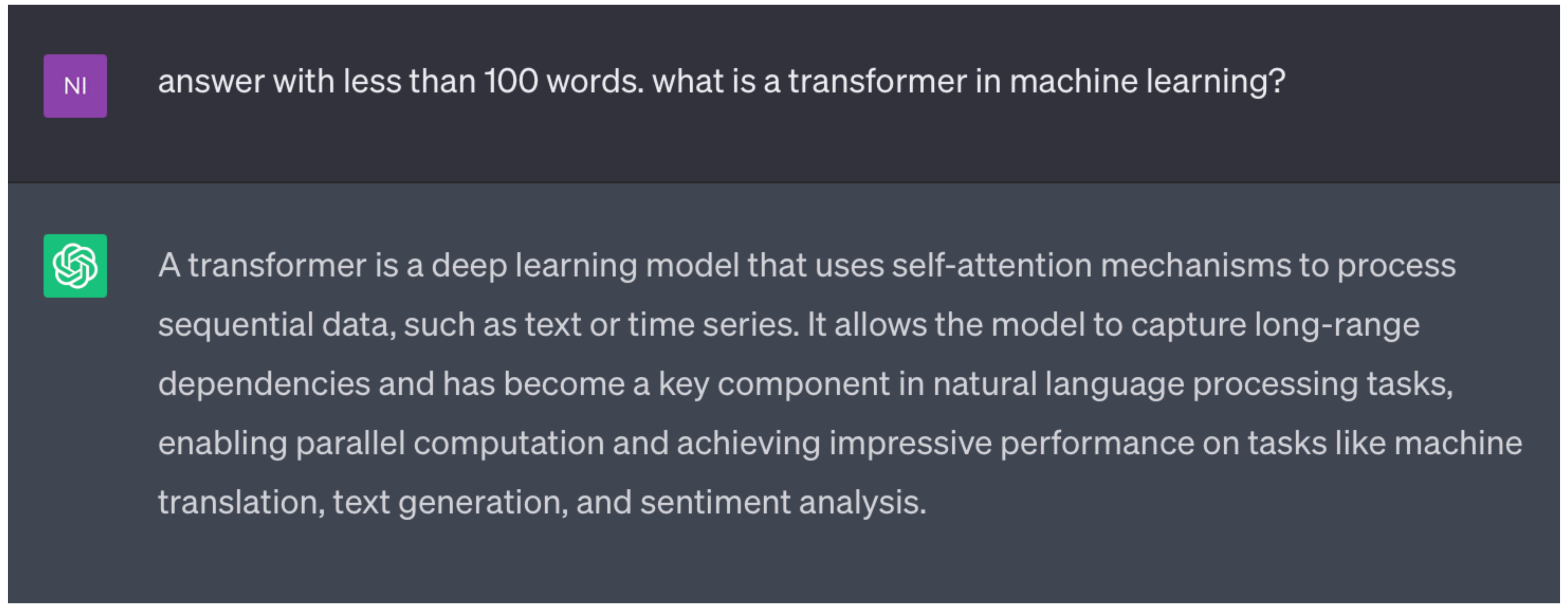
Stochasticity

Same questions, but two different answers!



[redacted] answer with less than 100 words. what is a transformer in machine learning?

A transformer is a type of deep learning model used in machine learning, particularly in natural language processing (NLP). It is based on a self-attention mechanism that allows the model to weigh the importance of different words or tokens in a sequence. Transformers have revolutionized NLP tasks by enabling parallel computation, capturing long-range dependencies, and achieving state-of-the-art performance on various tasks, such as machine translation, text generation, and sentiment analysis. They have become a fundamental building block in many modern NLP architectures.



NI answer with less than 100 words. what is a transformer in machine learning?

A transformer is a deep learning model that uses self-attention mechanisms to process sequential data, such as text or time series. It allows the model to capture long-range dependencies and has become a key component in natural language processing tasks, enabling parallel computation and achieving impressive performance on tasks like machine translation, text generation, and sentiment analysis.

The model is quite deterministic. Why do the answers change?

→ **Sampling**

Stochasticity

- For each position, the network predicts a probability vector
  - Probability to predict each token
  - *Classification problem, remember?*
  - Then we take argmax, and get the token with highest probability

- Instead of argmax, we can sample!
- For example,
  - Let’s pick the 5 tokens with the highest probabilities
  - Randomly pick one token (sampling)
- *Deterministic network, stochastic outputs*

Some additional interesting info about chatGPT and family

We won't implement them

Autoregressive model

- chatGPT is an autoregressive model
  - Predicts one token at a time
  - *Attention will look into **prompt** and the **previous tokens***

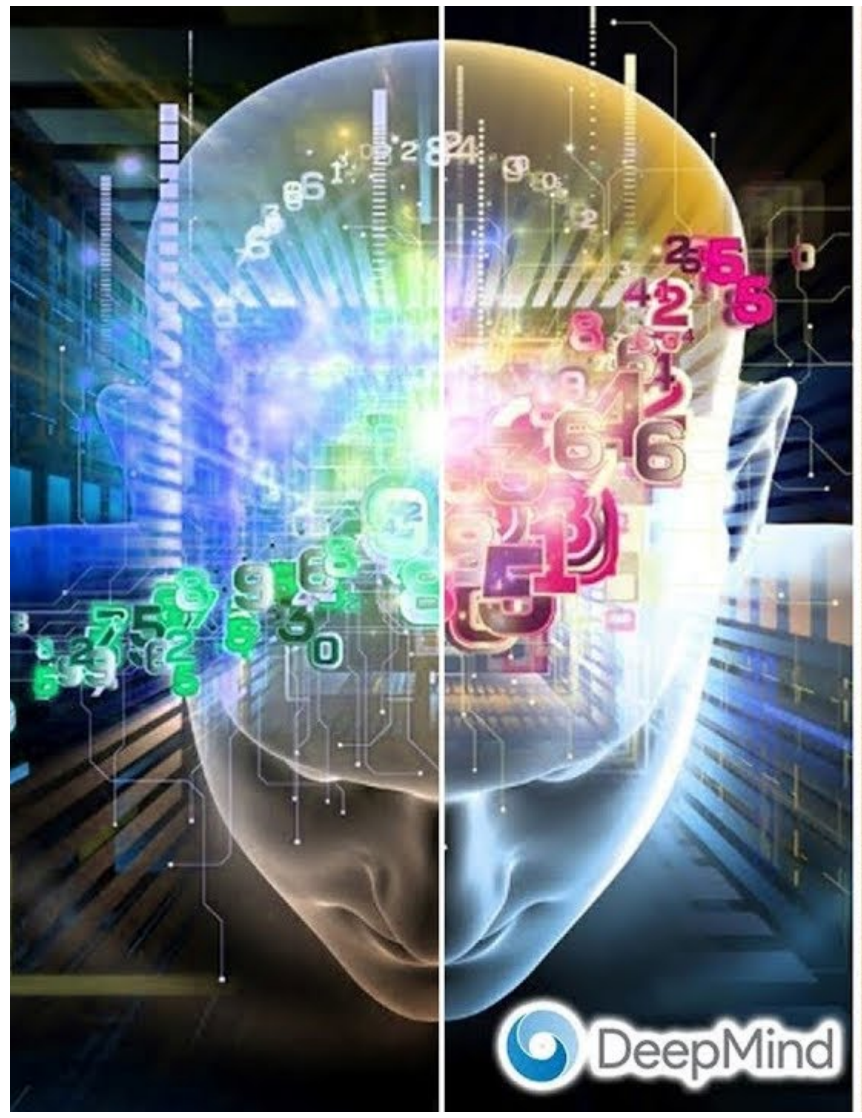
- Quite advantageous over what we did (constant length)
  - Most common approach in these Large Language Models (LLMs)

- GPT3 was mostly trained on common crawl dataset
  - “Data from the internet”
  - People on the internet are not always very nice
    - Internet is full of discriminatory/abusive language

- Yet, chatGPT is so humble, polite, polished, full of positivity
  - How??

- *Reinforcement learning*

Reinforcement learning



- A type of ML
  - Your actions have consequences!
  - The network (agent) interact with an environment
    - Playing chess, video game
    - Self-driving car
  - Network output (action) will interact and change environment, based on the changed environment, network will predict the next action
    - Constant feedback loop
  - Network gets reward if its actions help attain the final goal (win the chess match)
- The LLMs usually go through an RL stage after the supervised training
  - Gets rewarded for being nice!

Now you can train your own GPT

- You just need to
  - Get a Large dataset
  - Get ~1k GPUs
  - Build a network of trillions of parameters
  - Improve over everything we discussed by a looooot
  - Train for a really long time (~month)
  - Get a $100M funding (rumored training cost of GPT4)