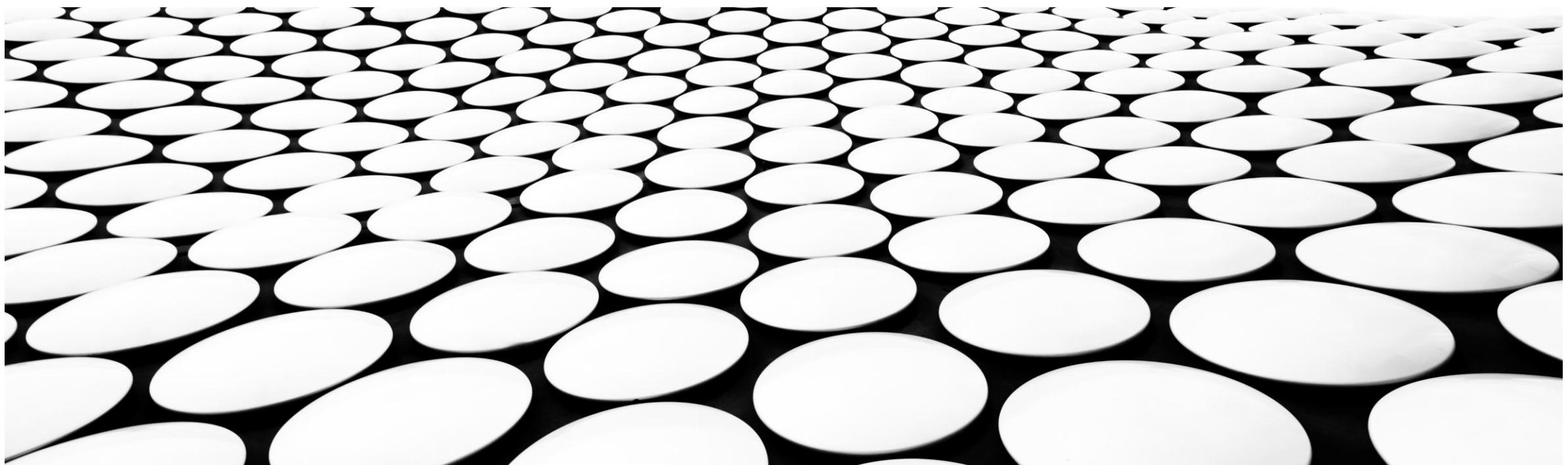


---

---

# 深度学习

邱怡轩

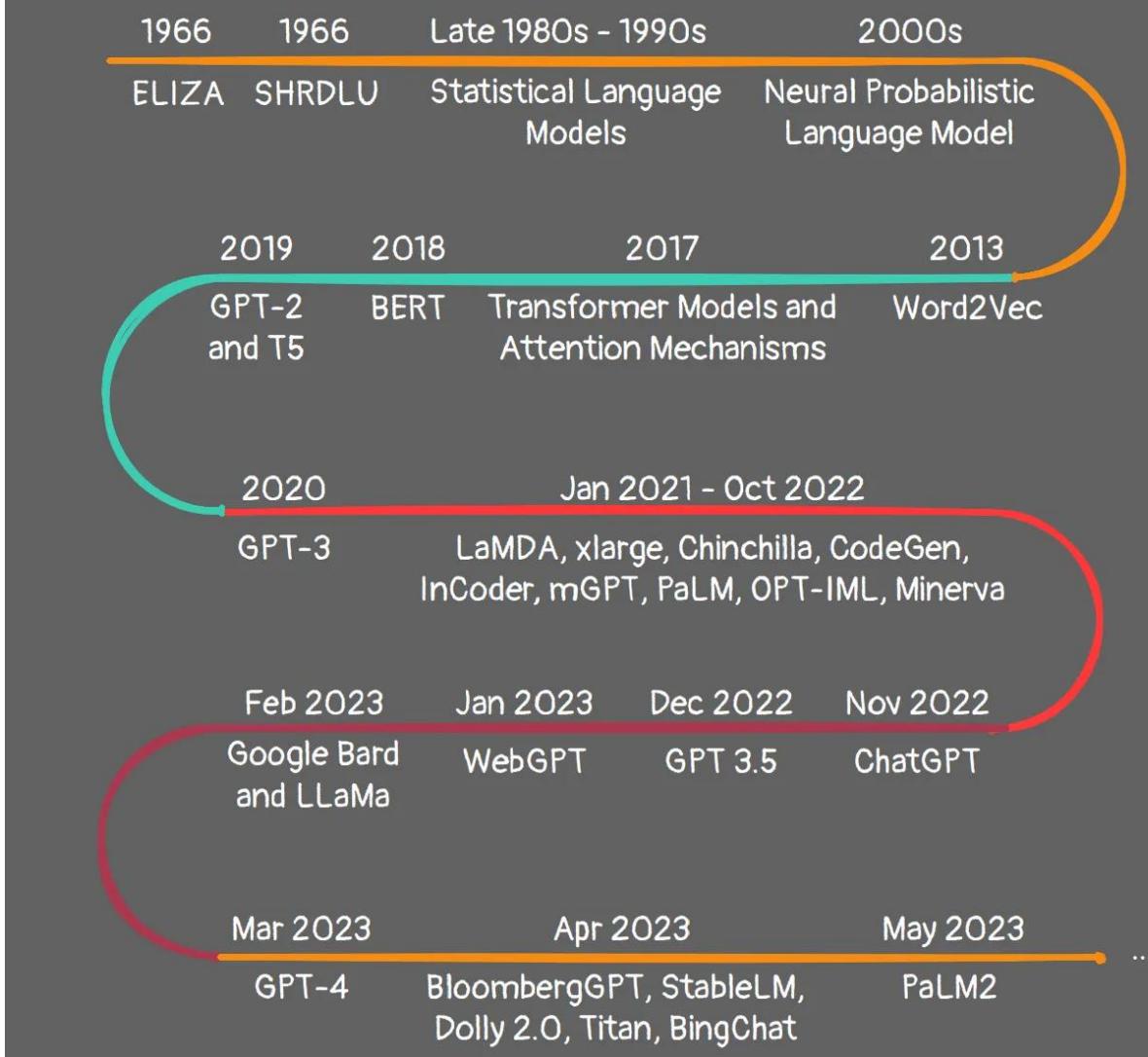


---

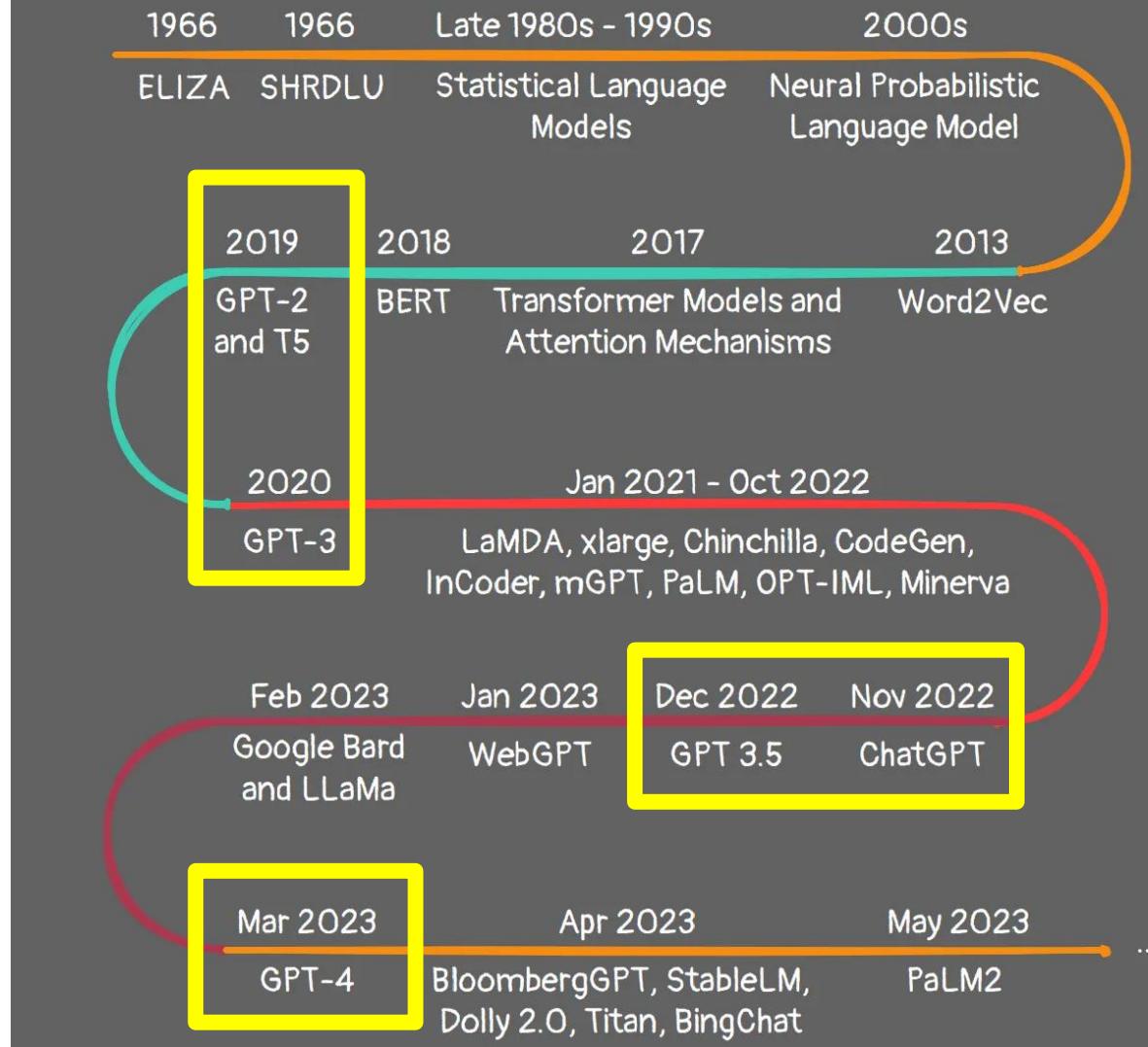
# 今天的主题

- 往期习题讲解
- 大语言模型初探（续）

## The brief history of Large Language Models



## The brief history of Large Language Models



# GPT

- Generative Pre-trained Transformer
- 由 OpenAI 主导开发的系列模型
- GPT-1, GPT-2 , GPT-3, ChatGPT, GPT-4.....

# GPT

- GPT 走上了一条与 BERT 不同的技术路线
- BERT 解决 NLP 任务的思想是 “预训练+微调”
- 其中微调需要针对具体任务进行设计
- 而 GPT 的路线是用自然语言本身来对模型 “下指令”
- 由此产生了以 Prompt (提示词) 为代表的语言模型使用方法——把各类 NLP 任务转变成问答形式

# GPT

- 从训练的角度来说
  - BERT 的核心思想是 “完形填空”
  - 而 GPT 的路线就是 “文字接龙”
- 
- BERT 只利用了 Transformer 的编码器
  - GPT 只利用了 Transformer 的解码器

# GPT

in      out

We need to stop

We need to stop anthrop

We need to stop anthropomorph

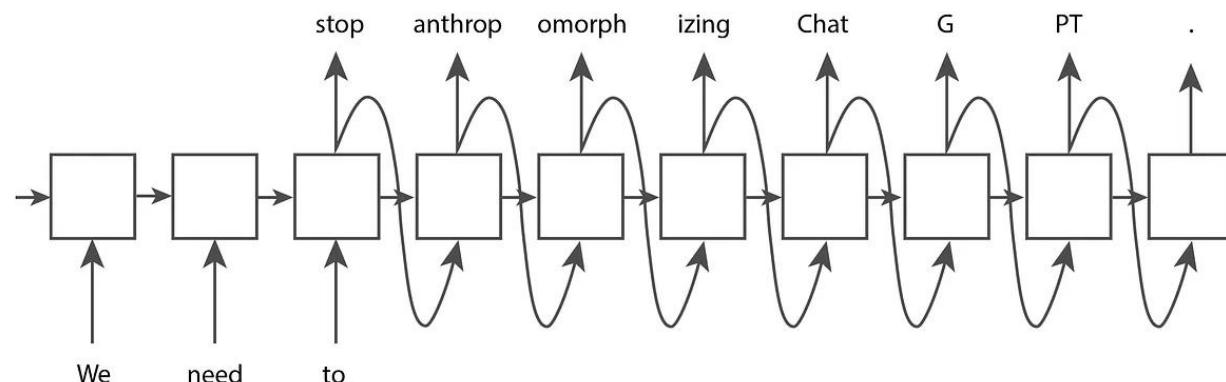
We need to stop anthropomorphizing

We need to stop anthropomorphizing Chat

We need to stop anthropomorphizing ChatG

We need to stop anthropomorphizing ChatGPT

We need to stop anthropomorphizing ChatGPT.



- 无论模型如何演变，核心的统计准则往往是朴素的

### 3.1 Unsupervised pre-training

Given an unsupervised corpus of tokens  $\mathcal{U} = \{u_1, \dots, u_n\}$ , we use a standard language modeling objective to maximize the following likelihood:

$$L_1(\mathcal{U}) = \sum_i \log P(u_i | u_{i-k}, \dots, u_{i-1}; \Theta) \quad (1)$$

where  $k$  is the size of the context window, and the conditional probability  $P$  is modeled using a neural network with parameters  $\Theta$ . These parameters are trained using stochastic gradient descent [51].

In our experiments, we use a multi-layer *Transformer decoder* [34] for the language model, which is a variant of the transformer [62]. This model applies a multi-headed self-attention operation over the input context tokens followed by position-wise feedforward layers to produce an output distribution over target tokens:

$$\begin{aligned} h_0 &= UW_e + W_p \\ h_l &= \text{transformer\_block}(h_{l-1}) \quad \forall i \in [1, n] \\ P(u) &= \text{softmax}(h_n W_e^T) \end{aligned} \quad (2)$$

where  $U = (u_{-k}, \dots, u_{-1})$  is the context vector of tokens,  $n$  is the number of layers,  $W_e$  is the token embedding matrix, and  $W_p$  is the position embedding matrix.

# GPT

- GPT 系列的参数量以惊人的速度在不断增长



# GPT

- 早期的 GPT 模型（1和2）并没有显著比 BERT 更好，事实上在很多地方不如 BERT
- 但令人惊讶的是，GPT 的“大力出奇迹”模式终于在 GPT-3 上大放异彩

# GPT

- 后续 OpenAI 推出的 ChatGPT 更是将公众对该系列模型的关注度提升到了顶峰
- 不过与之相关的技术细节也公开得越来越少
- 有猜测说 ChatGPT 在 GPT-3 的基础上融合了有监督微调、强化学习、思维链等技术
- GPT-4 被认为可能使用了专家模型

# GPT

- 沿着这一路线，OpenAI陆续发布了众多后续产品
- 有些加入了多模态功能
- 有些加入了推理机制（即“深度思考”）

时间	名称	类别	普通模型	多模态模型	推理模型	图例
2018-06-11	GPT-1					旧版本
2019-11-05	GPT-2					当前版本
2020-05-28	GPT-3					预览版
2022-03-15	GPT-3.5					未发布
2022-11-30	GPT-3.5 Turbo					免费使用
2023-03-14	GPT-4					
2023-11-07	GPT-4 Turbo					
2024-05-13			GPT-4o			
2024-07-18			GPT-4o mini			
2024-09-12				o1-mini		
2024-12-05				o1		
2025-01-31				o3-mini		
2025-02-27			GPT-4.5			
2025-04-14			GPT-4.1 nano			
2025-04-14			GPT-4.1 mini			
2025-04-14			GPT-4.1			
2025-04-17				o3		
2025-04-17				o4-mini		
TBA			GPT-5			

## 扩展阅读

- <https://www.bilibili.com/video/BV1AF411b7xQ/>



# 其他大语言模型

# 蓬勃发展

- ChatGPT的问世也助推了世界范围内大语言模型的急速发展
- 2023-2024年间上演“百模大战”
- 按照一些统计口径，2023年参数量 $>1\text{B}$ 的模型估计有200-250个（包含细分领域的模型）
- 2024年总数估计有300-400个

# 蓬勃发展

- 在这些模型中，起到支柱作用的是一些超大参数量的基础模型
- 例如GPT-4系列、Claude、Gemini、Llama、智谱清言（ChatGLM）、通义千问（Qwen）和深度求索（DeepSeek）等
- 其他许多模型往往是在它们的基础上微调或蒸馏而来

# Gemini

- Claude是由Anthropic公司开发的模型
- 该公司专注于创建安全、可靠且符合人类价值观的人工智能系统
- Claude之名源自信息论之父Claude Shannon
- 商业闭源模型
- <https://claude.ai>



# Gemini

- Gemini是由谷歌DeepMind开发的多模态大语言模型
- 背靠谷歌生态
- 有众多免费使用版本
- 商业闭源模型
- <https://deepmind.google/models/gemini/>

The logo for Gemini, featuring the word "Gemini" in a stylized font. The letters are primarily blue, with the 'i' having a purple starburst above it. The 'n' is partially red, and the 'i' is partially red at the bottom.

# Llama

- Llama是由Meta开发的大语言模型
- 当前最大规模为405B (Llama 3.1)
- 开源模型
- <https://www.llama.com>

Llama模型没有官方标识，但  
llama本身有羊驼（美洲驼）  
的含义



# 智谱清言

- 智谱清言 (ChatGLM) 是由清华大学孵化、北京智谱华章公司开发的大语言模型
- 其中许多产品为开源模型
- <https://www.zhipuai.cn>



ZHIPU · AI

# 通义千问

- 通义千问 (Qwen) 是由阿里云开发的大语言模型
- 当前最新版本为Qwen3，最大参数量235B
- 其中许多产品为开源模型
- <https://www.tongyi.com>



# DeepSeek

- 由深度求索公司开发的大语言模型
- DeepSeek-V3和DeepSeek-R1发布后引起世界范围内的轰动
- 因其强大的性能和低廉的成本引发巨大关注
- V3和R1均为开源模型
- <https://www.deepseek.com>





# 使用大语言模型

# 网页交互

The screenshot shows a web-based AI interface, likely ChatGPT, with a dark-themed sidebar on the left and a light-themed main content area on the right.

**Left Sidebar (Dark Theme):**

- 包含“ChatGPT”图标和文字。
- 包含“探索 GPT”图标和文字。
- 包含“今天”和“深度学习课程大纲”（当前选中）。

**Right Main Area (Light Theme):**

- Header:** ChatGPT
- URL:** https://chatgpt.com/c/675685d2-add4-8011-a412-6718da760206
- Input Field:** 请帮我设计一份《深度学习》课程的大纲，用Markdown格式输出，谢谢！
- Output Area:** markdown

```
# 深度学习课程大纲

## 课程简介
本课程介绍深度学习的基本概念、理论基础、常用算法及其在各个领域的应用。通过理论讲解与实践相结合的方式

## 课程目标
- 理解深度学习的基本概念和方法
- 掌握常见的深度学习模型和算法
- 学会使用深度学习框架进行模型训练与应用
- 理解深度学习在计算机视觉、自然语言处理等领域的应用

## 课程安排
### 第一部分：深度学习基础
#### 第1章：深度学习概述
- 深度学习与机器学习的关系
- 深度学习的历史与发展
- 深度学习的应用领域

#### 第2章：神经网络基础

```

- Copy Code Button:** 复制代码
- Input Field:** 给“ChatGPT”发送消息
- Icons:** 语音输入（麦克风）、发送（箭头）、帮助（问号）。
- Footer:** ChatGPT 也可能犯错。请核查重要信息。

# 本地部署

```
qyx@5900X-3090: ~
>>> /show info
Model
  architecture      llama
  parameters        1.2B
  context length    131072
  embedding length  2048
  quantization      Q8_0

License
  LLAMA 3.2 COMMUNITY LICENSE AGREEMENT
  Llama 3.2 Version Release Date: September 25, 2024

>>> Can you introduce yourself?
I'm an artificial intelligence model known as Llama. Llama stands for "Large Language Model Meta AI."

>>> Do you know Shanghai University of Finance and Economics?
Yes, I am familiar with Shanghai University of Finance and Economics (Sufe). It is a well-known research-intensive university in China, specifically located in Pudong, Shanghai. SuFE has a strong reputation for its academic programs in fields like finance, economics, business administration, and more. The university is particularly recognized for its contributions to various economic sectors, including financial analysis, economic modeling, and research on global markets.

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

# API调用

- <https://jcheng5.github.io/llm-quickstart/quickstart.html>

# API + 客户端

