AI 分享

张驰

April 27, 2025

黑客帝国动画版（2003）海报

1 引子

近年，AI 对不少行业都形成了一定的影响，大家对 AI 的热情也愈发高涨。希望通过这篇文章以简单直观的方式介绍大模型领域的常见概念、各种时髦的名词及新兴技术（像是 AI Agent、Function Call、MCP、A2A 等），帮助大家了解当前的热点技术与工具，当工作中涉及 AI 时，可以有一定的判断。这篇文章主要讲解大模型相关技术与工具的背景、功能与使用场景，对于这些背后的原理只做简单阐述，如有兴趣可以进一步阅读参考文献。

2 大模型的工作原理

在阐述大模型相关的技术与工具前，需要大家首先对大模型的工作原理有一定了解。对于大模型的详细工作原理可参阅文献 [1]。其实大模型的工作原理很简单，就是四个字“文字接龙”，大模型根据输入的 Prompt 计算出下一个可能的字或词，再用此 Prompt 附加上新产生的字或词作为大模型新的输入，重复此过程直至得到完整的结果，大模型的简易工作原理如图1所示：

举个例子，如果向大模型输入 Prompt“今天天气怎么样?”，大模型大概会按以下流程进行工作并输出结果：

1. 分词，大模型首先对 Prompt 进行分词处理，将其分解为 token 序列“今天”、“天气”、“怎么样”与“?”。分词是自然语言处理中非常重要的一步，目的是将连续的文本按语义或语法规则

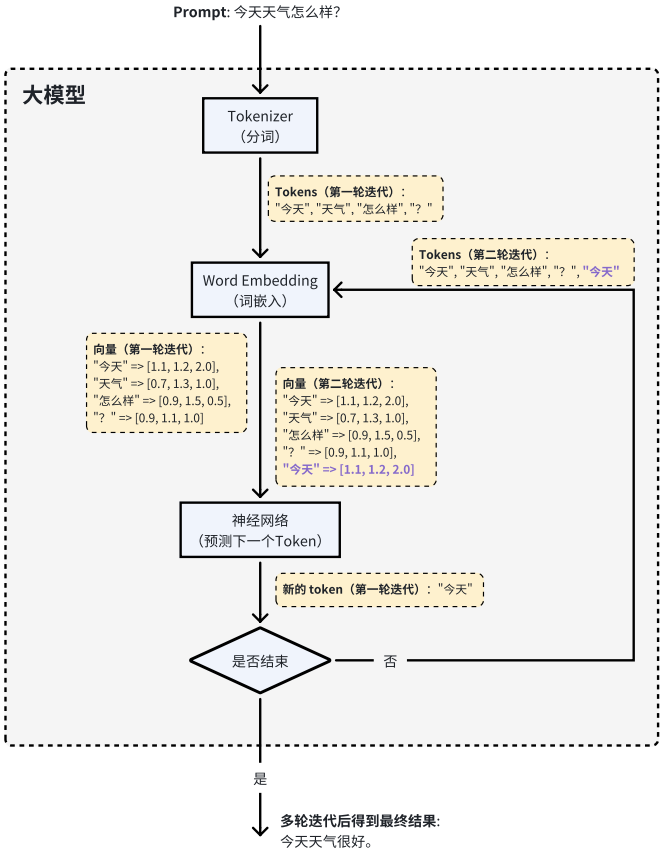

图 1: 大模型的工作流程

切分成独立的词语单元，最近大名鼎鼎的 DeepSeek V3 模型采用的是 Byte-level BPE 分词算法 [2]。另外，在进行分词前，通常需先对文本进行归一化（Normalize）处理，目的是将不同形式、书写习惯或字符表示的文本统一为一致的格式，以避免因表面形式的差异导致分词错误。对于分词的详细可参阅文献 [3]，另外 Hugging Face 也提供了一个 Tokenizer 的 Rust 实现 [4]。;

2. 词嵌入，大模型随后对 token 进行词嵌入处理，将每个 token 转化为一个向量，比如“今天”被转化为向量 $[1.1, 1.2, 2.0]$。这里大家可能会有疑问，这个功能是将 token 转化为向量，那为什么不叫作“向量化”，而是叫作“词嵌入”呢?“词嵌入”这个词最早出现于 Google AI 在 2013 年提出的 Word2vec 模型 [5]，词嵌入旨在将每个词赋予唯一的量化值，并且语义较近的词的量化值间的距离应当近，语义较远的词的量化值间的距离应当远，另外每个词在不同的上下文中可能会有不同的语义，为了承载语义的复杂性，量化值通常是多维向量，这里推荐阅读文献 [6];

3. 预测，大模型再将向量输入至神经网络进行计算得到一个新的 token，如“今天”;

4. 附加，大模型再将新的 token 附加至之前的 token 序列，得到新的 token 序列“今天”、“天气”、“怎么样”、“?”与“今天”;

5. 重复步骤 2-4 直至得到最终结果。

3 附录

3.1 Tranformer

$$q_i = h_i W_q, k_i = h_i W_k, v_i = h_i W_v \tag{3.1}$$

3.2 vLLM

vLLM[7]

大语言模型（LLM）的高吞吐量服务需要一次批处理足够多的请求。每个请求的 KV cache 是巨大的且动态伸缩。vLLM（1）KV cache 几乎无内存浪费；（2）跨请求和请求内 KV cache 灵活共享。

Parallel Sampling Beam Search

one can think of blocks as pages, tokens as bytes, and requests as processes.

引用

[1] J. Alammar. "The illustrated transformer." Accessed: 2025-04-03, Personal Blog. (2018), [Online]. Available: https://jalammar.github.io/illustrated-transformer/.

[2] Y. Shibata, T. Kida, S. Fukamachi, *et al.*, "Byte pair encoding: A text compression scheme that accelerates pattern matching," 1999.

[3] Glan 格蓝. "LLM 大语言模型之 Tokenization 分词方法（WordPiece, Byte-Pair Encoding (BPE), Byte-level BPE(BBPE) 原理及其代码实现）." Accessed: 2025-04-04, Personal Blog. (2023), [Online]. Available: https://zhuanlan.zhihu.com/p/652520262.

[4] [Online]. Available: https://github.com/huggingface/tokenizers (visited on 04/04/2025).

[5] T. Mikolov, K. Chen, G. Corrado, and J. Dean, *Efficient estimation of word representations in vector space*, 2013. arXiv: 1301.3781 [cs.CL]. [Online]. Available: https://arxiv.org/abs/1301.3781.

[6] 康斯坦丁. "没有思考过 Embedding，不足以谈 AI." Accessed: 2025-04-06, Personal Blog. (2023), [Online]. Available: https://zhuanlan.zhihu.com/p/643560252.

[7] W. Kwon, Z. Li, S. Zhuang, *et al.*, *Efficient memory management for large language model serving with pagedattention*, 2023. arXiv: 2309.06180 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2309.06180.