大语言模型（LLMs）：操作指南

翻译来自：喻四

介绍

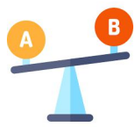
LLMs（语言模型）是人工智能模型，可以根据在大量训练数据中发现的模式生成类似人类的文本。它们用于语言翻译、聊天机器人和内容创作等应用。

一些热门 LLMs

这些流行的语言模型包括由**OpenAI**开发的**GPT-3**（Generative Pretrained Transformer）、由**Google** 开发的 **BERT**（Bidirectional Encoder Representations from Transformers）以及由**卡内基梅隆大学** 和 **Google** 共同开发的 **XLNet**（eXtreme MultiLingual Language Model）。

BigScience	BLOOM	176B	July 2022
	T0pp	11B	October 2021
EleutherAI	GPT-J	6B	July 2021
	GPT-NeoX	20B	February 2022
清华大学 Tsinghua University	GLM	130B	August 2022
Google Research	UL2	20B	October 2022
	T5	11B	February 2020
Meta AI	OPT	175B	June 2022
	OPT	66B	June 2022
Yandex	YaLM	100B	June 2022

如何选择 LLMs



在比较不同的模型时，重要的是要考虑它们的**架构、模型大小、使用的训练数据量**以及它们在特定自然语言处理（NLP）任务上的性能。

LLMs（语言模型）的组成

LLMs通常由**编码器、解码器和注意力机制**组成。编码器接收输入文本并将其转换为一组隐藏表示，而**解码器生成输出**文本。**注意力机制**帮助模型集中关注输入文本中最相关的部分。

LLMs（语言模型）的应用

- LLMs被广泛应用于各种领域的应用，包括**语言翻译、聊天机器人、内容创作和文本摘要**等。
- 它们还可用于改进**搜索引擎、语音助手和虚拟助手**。

怎么训练语言模型（LLMs）

LLMs 使用一种称为**无监督学习**的过程进行训练。这涉及将大量的文本数据（如书籍、文章和网站）**输入模型**，并让模型学习文本中单词和短语之间的模式和关系。然后，模型根据特定**任务进行微调**，如语言翻译或文本摘要。

预处理

文本格式化，将文本转换为标准格式的过程，例如将所有文本转换为小写，删除特殊字符，并将数字转换为其书面形式。

分词，将文本分解为个别单元（例如单词或短语）的过程。这是准备文本数据进行自然语言处理任务的重要步骤。

停用词，通常在文本处理过程中删除的常见词汇，因为它们没有太多意义，并且可能引入噪音或影响自然语言处理任务的结果。停用词的例子包括“the”、“a”、“an”、“in”和“is”。

词形归并，根据词性和上下文将词语还原为其基本形式的过程。它是一种比词干提取更复杂的技术，能够产生更准确的结果，但在计算上更加昂贵。

词干提取和词形归并，将词语还原为其基本形式的技术。这有助于减少数据的维度，并提高模型的性能。

Fine-Tuning（微调）

微调（Fine-tuning）是指使用较小的数据集对预训练的大型语言模型进行**特定任务的训练过程**。这使得模型能够学习任务特定的特征并**提高性能**。

微调的过程通常涉及**冻结预训练模型的权重**，仅**训练特定任务的层**。

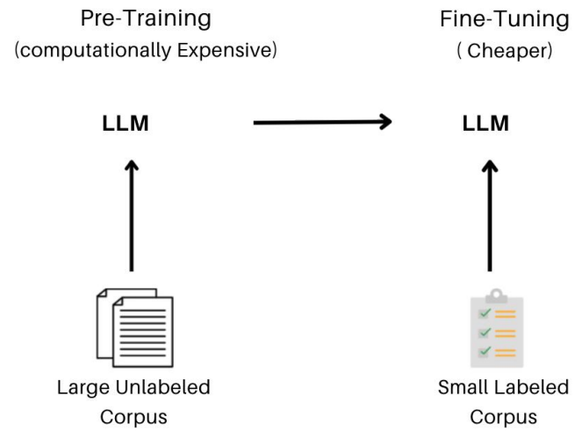
在**微调模型时**，重要的是**考虑**微调**数据集的大小、优化器和学习率**的选择以及评估指标的选择等因素。

微调语言模型 LLMs 的例子

- **模型成本：**每月500美元至5000美元不等，具体取决于语言模型的大小和复杂性。
- **GPU规格：**NVIDIA GeForce RTX 3080或更高。
- **GPU数量：**1-4个，具体取决于语言模型的大小和所需的微调速度。例如，微调GPT-3模型（这是目前最大的语言模型之一）至少需要4个GPU。
- **GPT-3：**进行微调的数据大小可以根据具体用例和模型本身的大小而有很大的差异。GPT-3是目前最大的语言模型之一，具有超过**1750亿**个参数，因此通常需要大量数据进行微调，以实现显著的性能改进。

需要注意的是，仅使用几十GB的小型数据集对GPT-3进行微调可能不会在性能上有显著的改进，而使用几TB规模的更大数据集进行微调可能会带来实质性的改进。微调数据的大小还取决于要微调的具体NLP任务和所需的精确度水平。

这只是一个例子，实际成本和GPU规格可能因语言模型、微调数据和其他因素而有所不同。最好向语言模型提供者咨询以获取最新信息和关于微调的具体建议。



输入表示
Input Representations

- **词嵌入（Word embeddings）**：每个标记（token）被替换为一个向量，表示其在连续向量空间中的含义。常见的词嵌入方法包括Word2Vec、GloVe 和 fastText。
- **子词嵌入（Subword embeddings）**：每个标记被分解为较小的子词单元（例如字符或字符n-gram），并用表示其含义的向量来替换每个子词。这种方法可以处理未登录词（OOV）并提高模型捕捉形态和语义相似性的能力。常见的子词嵌入方法包括字节对编码（BPE）、一元语言模型（ULM）和SentencePiece。
- **位置编码（Positional encodings）**：由于LLMs操作标记序列，需要一种方法来编码每个标记在序列中的位置。位置编码是添加到词或子词嵌入中的向量，提供关于每个标记位置的信息。
- **段落编码（Segment embeddings）**：在某些LLMs中，如Transformer，输入序列可以分为多个段落（例如句子或段落）。段落编码被添加到词或子词嵌入中，表示每个标记所属的段落。

Ashish Patel • Principal Research Scientist • ashishpatel.ce.2011@gmail.com

Abonia Sojasingarayar • Machine Learning Scientist • aboniaa@gmail.com

Updated: 2023-02

大语言模型（LLMs）：操作指南

翻译来自：喻四

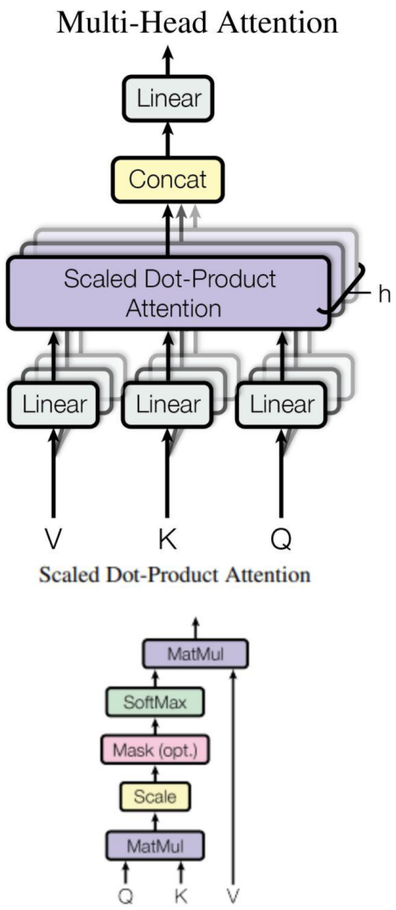
注意力机制 Attention Mechanisms

自注意力（Self-Attention）：

- 一种机制，允许序列在计算每个元素的表示时，权衡序列中所有其他元素的重要性。
- 能够捕捉序列中不同元素之间的关系，非**常适用于需要建模长距离依赖关系的任务**。
- **由于Transformer**模型广为流行。

多头注意力（Multi-Head Attention）：

- 自注意力的一种变体，其中注意力机制同时以不同的权重集合并行应用多次。
- 使模型能够关注输入序列的不同方面，**提高其捕捉复杂模式和依赖关系的能力**。
- 每个“头”产生一个单独的输出，这些输出被串联并进行线性转换以生成最终输出。
- 在**Transformer**模型中也常常使用。



语言模型（LLMs）评估

- **准确率（Accuracy）：**衡量正确分类的实例在所有实例中所占的比例。这个指标通常用于文本分类任务，比如情感分析，目标是正确地将文本分类为积极、消极或中性。
- **F1分数（F1-score）：**一个综合考虑精确率和召回率的指标。精确率是真正例在所有预测为正例的结果中的比例，而召回率是真正例在所有实际为正例的结果中的比例。F1分数是精确率和召回率的调和平均值，它提供了一个平衡的模型性能度量，适用于文本分类、问答等任务。
- **困惑度（Perplexity）：**它衡量语言模型在预测序列中的下一个词时的准确程度。困惑度越低，模型在预测下一个词方面的能力就越好。困惑度计算为2的交叉熵幂，交叉熵是衡量模型预测概率与序列中下一个词的真实概率相匹配程度的指标。此外，文本还可以使用BLEU或ROUGE分数等指标进行评估。

例子 - ChatGPT

ChatGPT 是由OpenAI开发的大型语言模型，它经过了在各种互联网文本（包括文章、论坛和社交媒体）上的训练。这种训练使得ChatGPT能够生成类似人类的文本，并回答许多不同主题的各种问题。

ChatGPT的一个用例

Question Answering: ChatGPT can answer factual questions based on the information it has been trained on. Example:

Human : What is the capital of France?
ChatGPT: The capital of France is Paris.

Conversational: ChatGPT can engage in a conversation with a user. Example:

Human : Hi, how are you today?
ChatGPT: Hello! I'm just an AI, so I don't have emotions, but I'm functioning well today. How can I assist you?

Text Generation: ChatGPT can generate text based on a prompt or seed text. Example:

Human : Write a short story about a magical world.
ChatGPT: Once upon a time, in a land far away, there was a magical world filled with wonder and enchantment. The sky was always a brilliant shade of purple, and the stars twinkled like diamonds in the sky. The people of this world possessed magical powers that they used to create incredible wonders and to help those in need. One day, a young woman discovered that she too had magical powers, and set out on an incredible journey to explore the wonders of this magical world and to hone her skills.

Summarization: ChatGPT can summarize a long text or article. Example:

Human : Please summarize this article about the benefits of exercise.
ChatGPT: The article discusses the various physical and mental health benefits of exercise, including improved cardiovascular health, stronger muscles and bones, better weight management, and reduced stress and anxiety. It also highlights the importance of finding an exercise routine that is enjoyable and sustainable.

Translation: ChatGPT can translate text from one language to another. Example:

Translate this sentence from English to Spanish. "The sky is blue."
ChatGPT: El cielo es azul.

语言模型（LLMs）的工具和库

a. 比较流行的自然语言处理（NLP）库，如**TensorFlow、PyTorch、spaCy、Hugging Face Transformers、AllenNLP、OpenAI GPT-3 API、AllenAI的ELMO、spaCy Transformers**等，提供了与大型语言模型一起工作的工具。这些库可以方便地进行模型的微调和部署。

b. 一些大型语言模型，如GPT-3，提供了API以访问其模型。这可以简化将大型语言模型集成到现实应用程序中的过程。

TensorFlow spaCy PyTorch OpenAI

语言模型（LLMs）面临的问题和挑战

- LLM模型面临的主要挑战之一是可能存在偏见或冒犯性语言，因为这些模型是从训练数据中学习到的模式中进行训练的。
- 伦理考虑，如性别和种族偏见。
- 训练和运行LLM所需的计算资源的数量，这可能会很昂贵且能源密集。
- 处理词汇表之外的词汇。
- 提高可解释性。虽然大型语言模型在各种NLP任务中表现出令人印象深刻的性能，但在某些需要更深入理解基础背景的任务中可能表现不佳。

语言模型（LLMs）的未来

LLMs的未来前景非常乐观，目前的研究重点是提高其准确性、减少偏见，并使其更易于访问和节能。

随着对以人工智能驱动的应用的需求不断增长，LLM将在塑造人机交互的未来中扮演越来越重要的角色。

Ashish Patel • Principal Research Scientist • ashishpatel.ce.2011@gmail.com

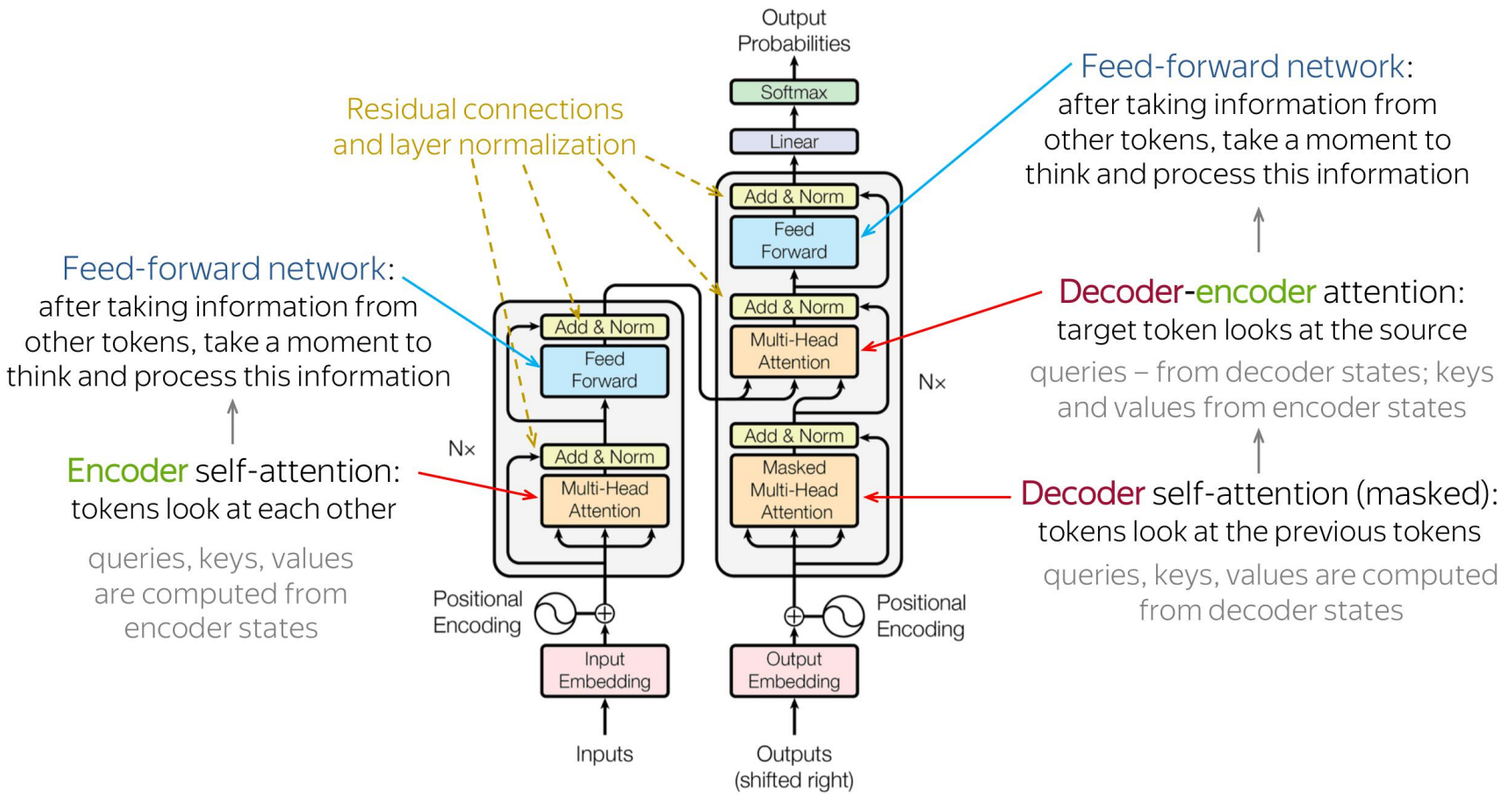
Abonia Sojasingarayar • Machine Learning Scientist • aboniaa@gmail.com

Updated: 2023-02

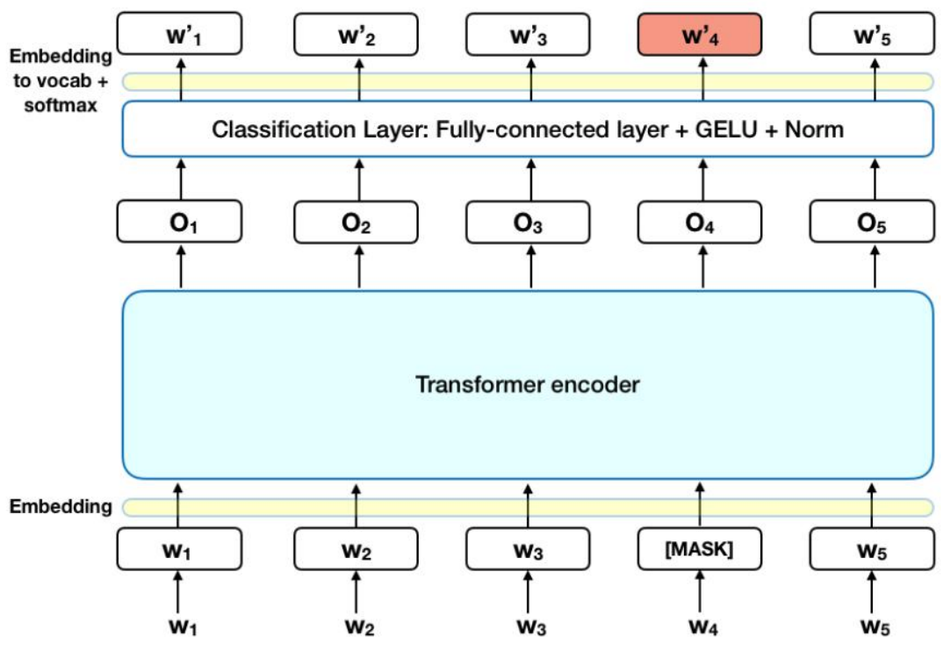
大语言模型（LLMs）：操作指南

翻译来自：喻四

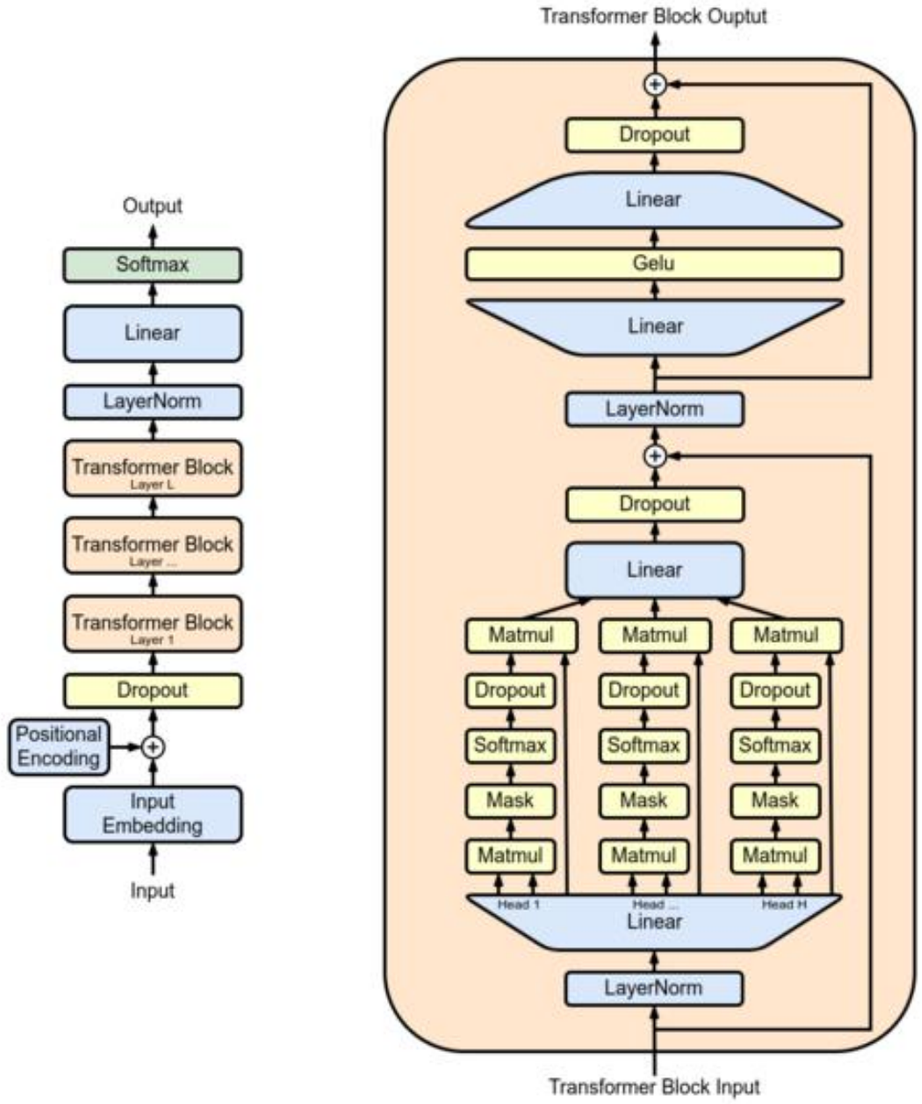
Transformer 架构



BERT 架构



GPT 架构



Ashish Patel • Principal Research Scientist • ashishpatel.ce.2011@gmail.com

Abonia Sojasingarayar • Machine Learning Scientist • aboniaa@gmail.com

Updated: 2023-02