LLaMA base model

> 在论文中，作者针对常识推理、问答、数学推理、代码生成、语言理解等能力对 LLaMA 进行了评测。结果显示，LLaMA 以相对少量的参数获得了媲美超大模型的效果，这对 NLP 社区的研究者们更加友好，因为它可以**在单个 GPU 上运行**。开源代码提供 LLaMA 的文本生成示例，可以直接用于一些 Zero/Few-Shot Learning 任务。也有许多用户关心如何使用自己的数据微调或增量训练 LLaMA 模型

背景介绍

「什么是 LLaMA：」

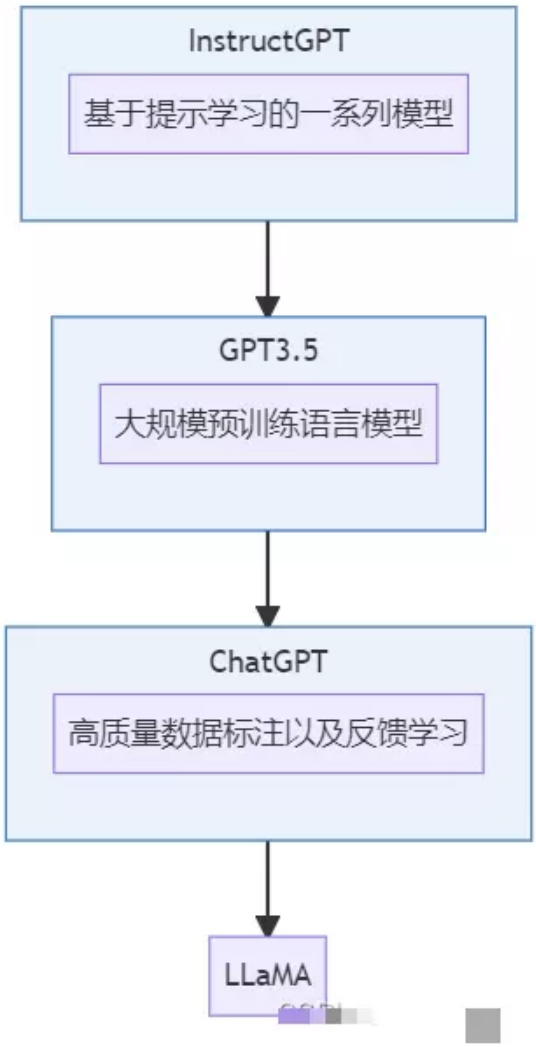
1. 参数量有四档：7/13/33/65 亿，最低档据24g 显存的显卡可以跑，7 亿的 LLaMA 用了 1 万亿 token 进行训练，最大模型则用了 1.4 万亿。
2. 用了万亿个 token 进行训练（所有数据均来自公开数据集）
3. 性能和 175 亿参数的 GPT-3 相当
4. 由 Meta AI 于 2023 年 2 月发布，作为致力于开放科学和人工智能实践的一部分

LLaMA 与其他大型语言模型的关联：」

LLaMA 与 GPT、GPT-3、Chinchilla 和 PaLM 等其他大型语言模型类似，因为它使用 Transformer architecture 来预测给定单词或 token 序列作为输入的下一个单词或 token。

LLaMA 与其他模型的不同之处在于，它使用了更多 token 进行训练，得到较小模型，这使它更高效，资源密集度更低。（可部署在 CPU 上做预测）

「LLaMA 发展史」



「LLaMA 的特点」

- 语种： LLaMA 涵盖了 20 种使用者最多的语言，重点是那些使用拉丁字母和西里尔字母的语言。这些语言包括英语、西班牙语、法语、俄语、阿拉伯语、印地语、汉语等。
- 生成方式：和 GPT 一样
- 所需资源更小： LLaMA 比其他模型更高效，资源密集度更低，因为它使用在更多 tokens 上训练的较小模型。这意味着它需要更少的计算能力和资源来训练和运行这些模型，也需要更少的内存和带宽来存储和传输它们。例如，LLaMA 13B 在大多数基准测试中都优于 GPT-3 175B），而只使用了约 7% 的参数。这个特点也为个人部署 LLaMA 提供了可能，让研究人员实现现更多的可访问性和个性化，并探索新的用例和应用程序。

数据集及处理

Dataset	Sampling prop.	Epochs	Disk size
CommonCrawl	67.0%	1.10	3.3 TB
C4	15.0%	1.06	783 GB
Github	4.5%	0.64	328 GB
Wikipedia	4.5%	2.45	83 GB
Books	4.5%	2.23	85 GB
ArXiv	2.5%	1.06	92 GB
StackExchange	2.0%	1.03	78 GB

Table 1: **Pre-training data.** Data mixtures used for pre-training, for each subset we list the sampling proportion, number of epochs performed on the subset when training on 1.4T tokens, and disk size. The pre-training runs on 1T tokens have the same sampling proportion

trick

- Normalize the input of each transformer sub–layer to improve training stability.
- Use SwiGLU instead of ReLU to improve performance.
- Use rotary embedding instead of absolute positioning to improve performance.

Pre-normalization

SwiGLU

Rotary Embeddings

Model	Size	Training data
LLaMA (base model)	7B, 13B, 33B, 65B	Various
Alpaca	7B, 13B	52k GPT-3 instructions
Vicuna	7B, 13B	70k ChatGPT conversations
Koala-distill	7B, 13B	117k cleaned ChatGPT conversations
GPT4-x-Alpaca	13B	20k GPT4 instructions
WizardML	7B	70k instructions synthesized with ChatGPT/GPT-3
OpenAssistant LLaMA	13B, 30B	600k human interactions (OpenAssistant Conversations)

Model comparison

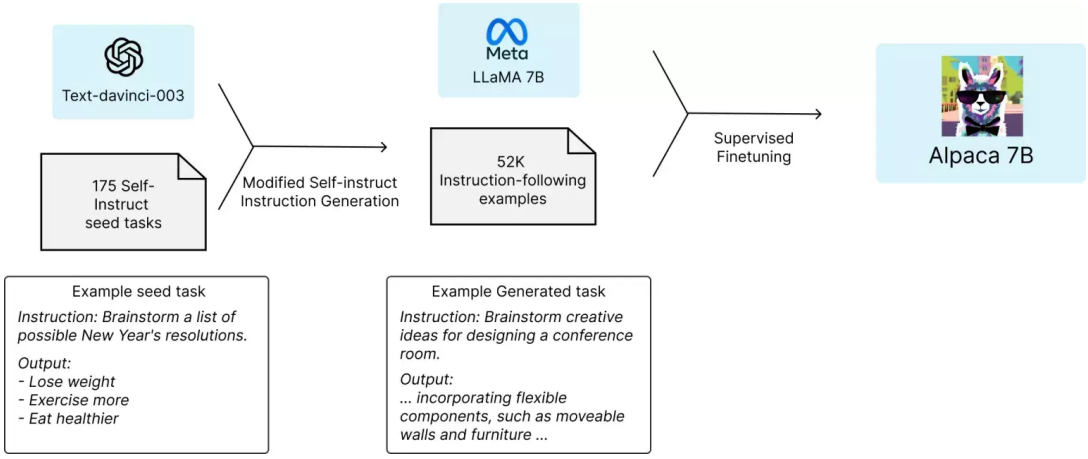
The table below summarizes the model parameters.

	Parameters	Layers	Attention heads	Embedding dimension
7B	6.7B	32	32	4,096
13B	13B	40	40	5,120
33B	33B	60	52	6,656
65B	65B	80	64	8,192

Alpaca model

Alpaca is a fine-tuned LLaMA model, meaning that the model architecture is the same, but the weights are slightly different. It is aimed at resolving the lack of instruction-following capability of LLaMA models.
It behaves like ChatGPT and can follow conversations and instructions.

The authors first generate the training data using OpenAI's GPT-3, then convert them to 52k instruction-following conversational data using the Self-Instruct pipeline.



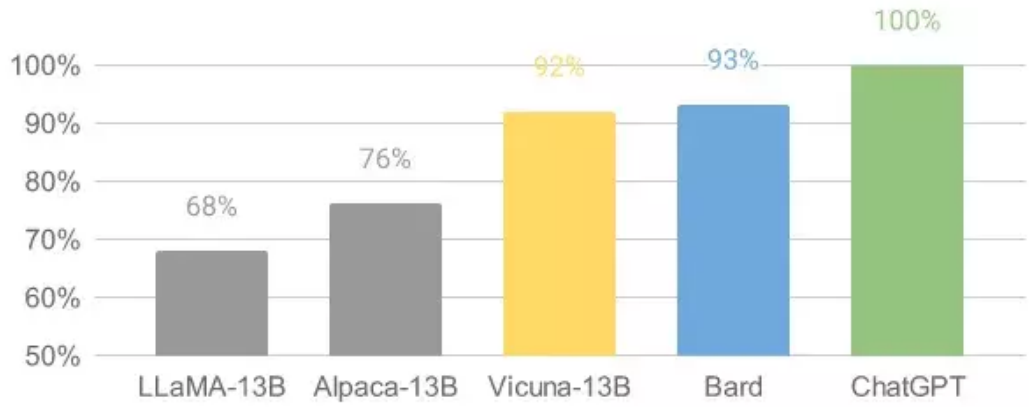
Vicuna model

The model was fine-tuned by an academic team from UC Berkeley, CMU, Stanford, and UC San Diego.

- Vicuna model page
- Installation Guide on Mac.

Vicuna is trained by fine-tuning the LLaMA base models on user-shared conversations collected from ShareGPT.com. So it is basically fine-tuned with ChatGPT conversations.
It comes in two sizes: 7B and 13B.

How good is Vicuna? According to their website, the output quality (as judged by GPT-4...) is about 90% of ChatGPT, making it the best language model you can run locally.



The authors used an interesting method to evaluate the model's performance: Using GPT-4 as the judge. They asked GPT-4 to generate some challenging questions and let Vicuna and some other best language models answer them.

They then ask GPT-4 to evaluate the quality of the answers in different aspects, such as helpfulness and accuracy.

The Vicuna model is **considered to be one of the best LLaMA models that you can** run locally.

Koala

Koala is a LLaMA 7B and 13B models fine-tuned with publicly available dialog data by an academic team at UC Berkeley.

The training data includes filtered data from multiple datasets.

- ShareGPT — 30k
- Human ChatGPT Comparison Corpus — 87k
- Open Instruction Generalist — 30k
- Stanford Alpaca (Training dataset for Alpaca) — 52k
- Anthropic HH — 50k
- OpenAI WebGPT — 20k
- OpenAI summarization — 93k

They trained two models

1. **Koala-All**: Used all datasets
2. **Koala-Distill**: Used the first two datasets (i.e., data distilled from ChatGPT)