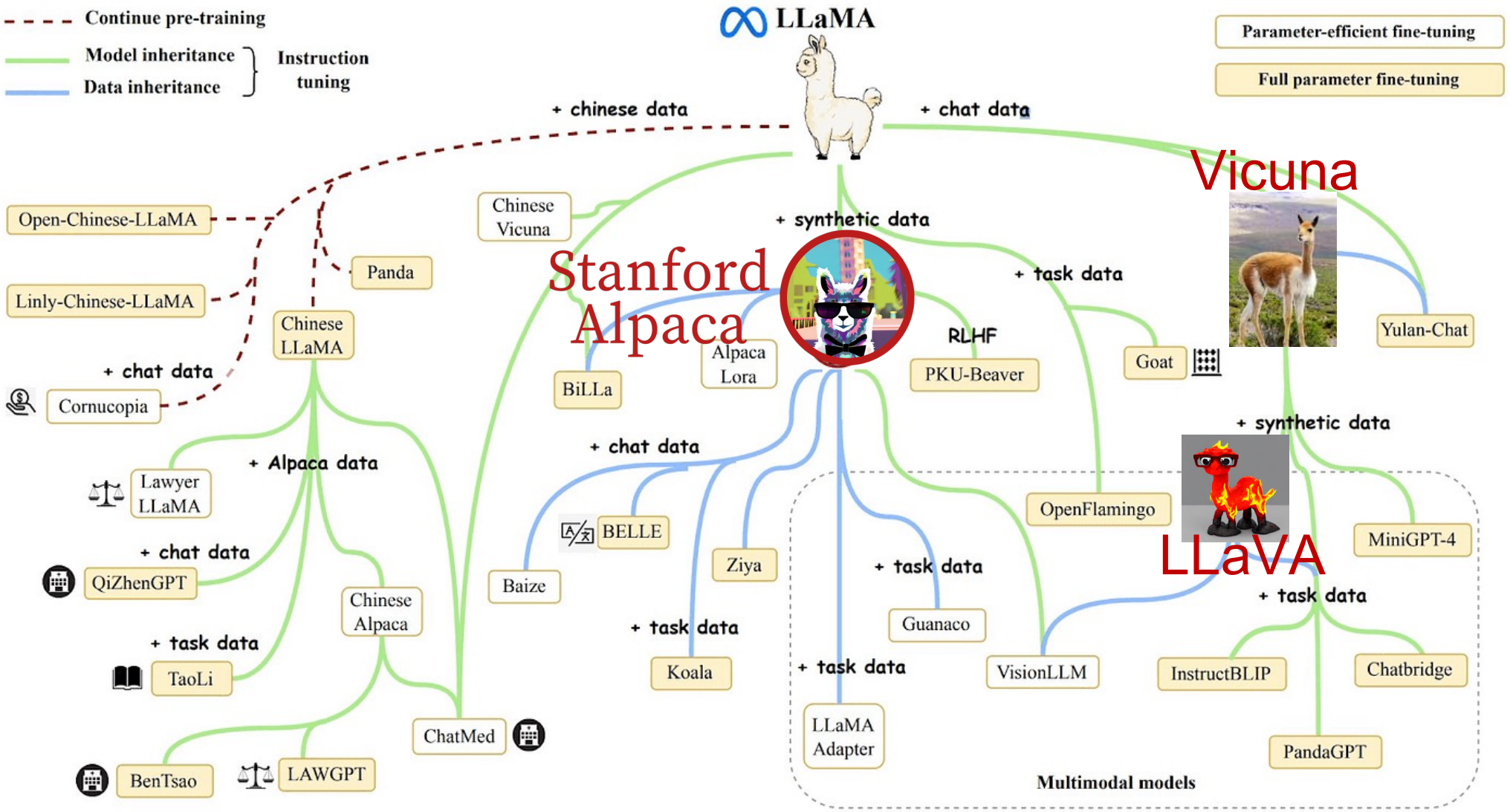
# LLaMa Family

# Alpaca, Vicuna, LLaVA

**LLM Reading Group**
**Houston Machine Learning Meetup**
**April 26, 2024**

# LLaMA Family




Ref: https://blackbearlabs.ai/blog-detail/open-large-language-models-history-2023-report

# Overview: LLAMA 1, 2 &3

| | Training Data | Params | Context Length | GQA | Tokens | LR |
|---|---|---|---|---|---|---|
| LLAMA 1 | *See Touvron et al. (2023)* | 7B | 2k | ✗ | 1.0T | $3.0 \times 10^{-4}$ |
| | | 13B | 2k | ✗ | 1.0T | $3.0 \times 10^{-4}$ |
| | | 33B | 2k | ✗ | 1.4T | $1.5 \times 10^{-4}$ |
| | | 65B | 2k | ✗ | 1.4T | $1.5 \times 10^{-4}$ |
| LLAMA 2 | *A new mix of publicly available online data* | 7B | 4k | ✗ | 2.0T | $3.0 \times 10^{-4}$ |
| | | 13B | 4k | ✗ | 2.0T | $3.0 \times 10^{-4}$ |
| | | 34B | 4k | ✓ | 2.0T | $1.5 \times 10^{-4}$ |
| | | 70B | 4k | ✓ | 2.0T | $1.5 \times 10^{-4}$ |

**Pretrained only**

**Pretrained + Instruction Fine-tuned (Llama2-chat)**

**Table 1: LLAMA 2 family of models.** Token counts refer to pretraining data only. All models are trained with a global batch-size of 4M tokens. Bigger models — 34B and 70B — use Grouped-Query Attention (GQA) for improved inference scalability.

LLAMA 3 8B, 70B, 400B+. 8k 15T

pretrained and instruction-fine-tuned

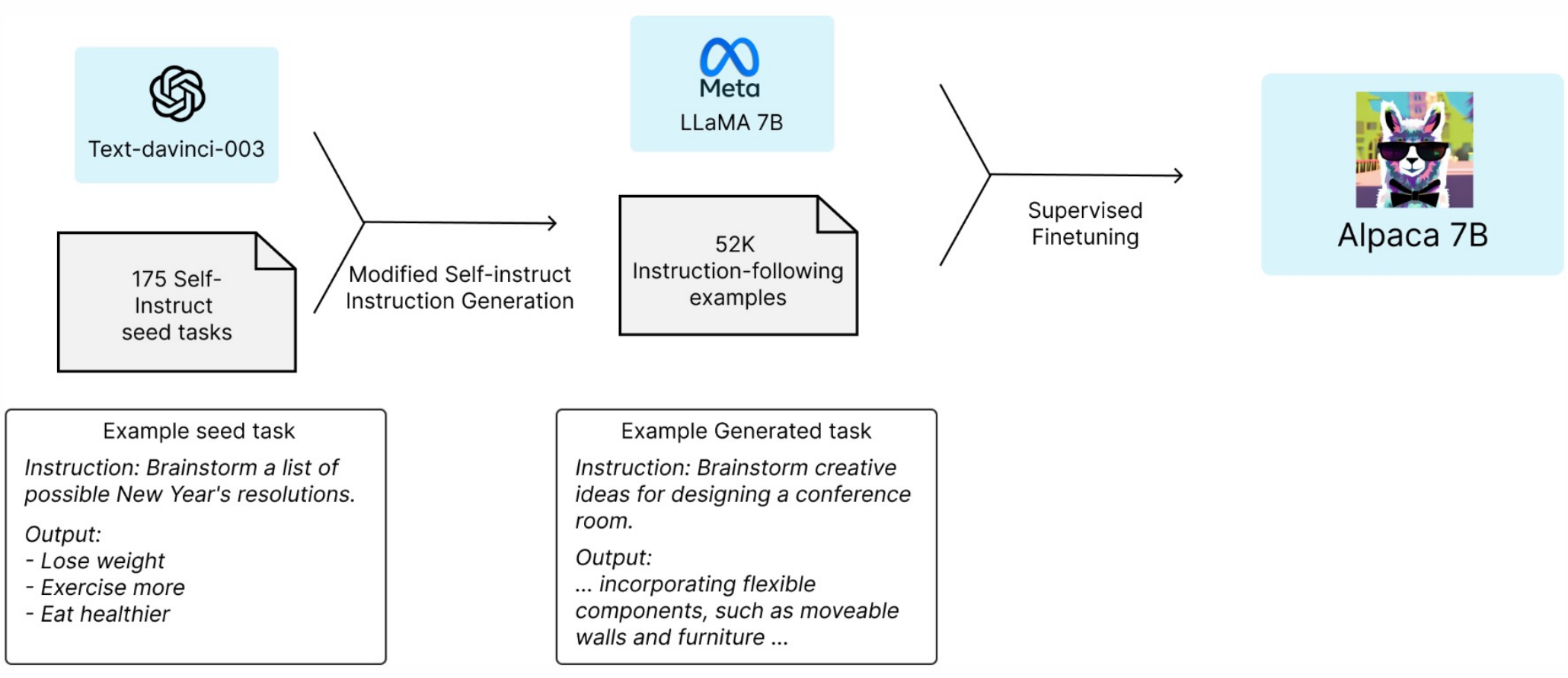
# Alpaca: A Strong, Replicable Instruction-Following Model


Released Mar 13, 2023

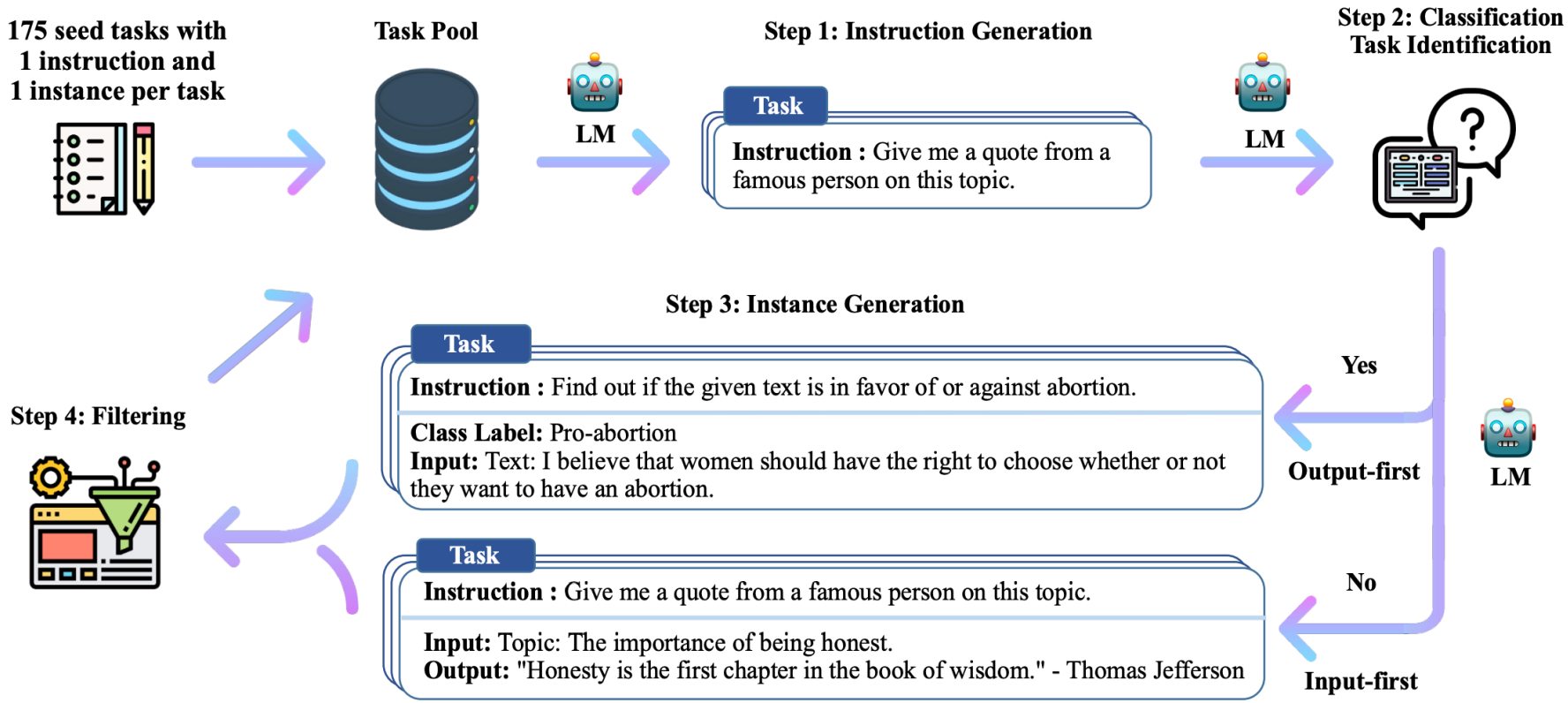
There are two important challenges to training a **high-quality instruction-following model under an academic budget**:

- A strong pretrained language model: Llama
- High-quality instruction-following data: self-instruct with a strong LLM

# Alpaca: Framework



# Self-Instruct: Aligning Language Models with Self-Generated Instructions



# Step 1. Instruction generation

```
Come up with a series of tasks:

Task 1:  {instruction for existing task 1}
Task 2:  {instruction for existing task 2}
Task 3:  {instruction for existing task 3}
Task 4:  {instruction for existing task 4}
Task 5:  {instruction for existing task 5}
Task 6:  {instruction for existing task 6}
Task 7:  {instruction for existing task 7}
Task 8:  {instruction for existing task 8}
Task 9:
```

Table 5: Prompt used for generating new instructions. 8 existing instructions are randomly sampled from the task pool for in-context demonstration. The model is allowed to generate instructions for new tasks, until it stops its generation, reaches its length limit or generates "Task 16" tokens.

# Step 2. Classification task identification

```
Can the following task be regarded as a classification task with finite output labels?

Task:  Given my personality and the job, tell me if I would be suitable.
Is it classification?  Yes

Task:  Give me an example of a time when you had to use your sense of humor.
Is it classification?  No

Task:  Replace the placeholders in the given text with appropriate named entities.
Is it classification?  No

Task:  Fact checking - tell me if the statement is true, false, or unknown, based on your
knowledge and common sense.
Is it classification?  Yes

Task:  Return the SSN number for the person.
Is it classification?  No

Task:  Detect if the Reddit thread contains hate speech.
Is it classification?  Yes

Task:  To make the pairs have the same analogy, write the fourth word.
Is it classification?  No

Task:  Given a set of numbers, find all possible subsets that sum to a given number.
Is it classification?  No

Task:  {instruction for the target task}
```

Table 6: Prompt used for classifying whether a task instruction is a classification task or not.

# Step 3. Instance generation

```
Come up with examples for the following tasks.  Try to generate multiple examples when possible.
If the task doesn't require additional input, you can generate the output directly.

Task:  Which exercises are best for reducing belly fat at home?
Output:
- Lying Leg Raises
- Leg In And Out
- Plank
- Side Plank
- Sit-ups

Task:  Converting 85 F to Celsius.
Output:  85°F = 29.44°C

Task:  Sort the given list ascendingly.
Example 1
List:  [10, 92, 2, 5, -4, 92, 5, 101]
Output:  [-4, 2, 5, 5, 10, 92, 92, 101]
Example 2
Input 2 - List:  [9.99, 10, -5, -1000, 5e6, 999]
Output:  [-1000, -5, 9.99, 10, 999, 5e6]

Task:  {Instruction for the target task}
```

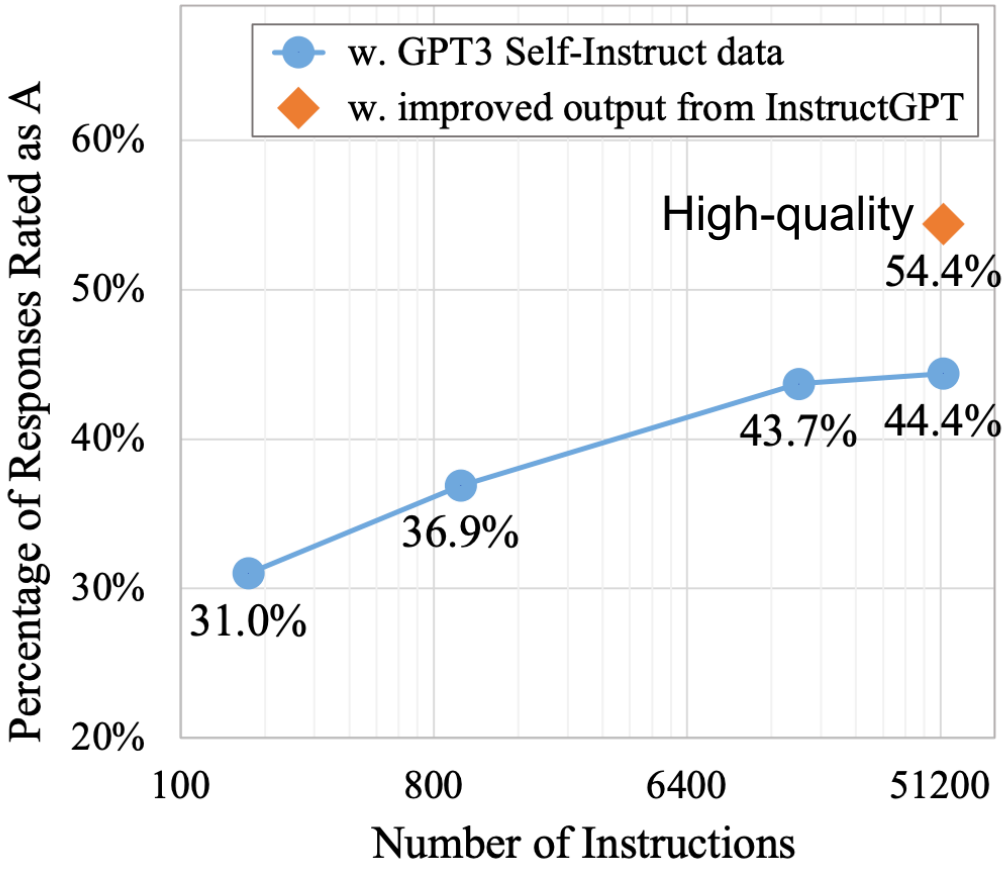
Table 7: Prompt used for the input-first approach of instance generation. The model is prompted to generate the instance first, and then generate the corresponding output. For instructions that don't require additional input, the output is allowed to be generated directly.

# Step 3. Instance generation

```
Given the classification task definition and the class labels, generate an input that
corresponds to each of the class labels.  If the task doesn't require input, just generate the
correct class label.

Task:  Classify the sentiment of the sentence into positive, negative, or mixed.
Class label:  mixed
Sentence:  I enjoy the flavor of the restaurant but their service is too slow.
Class label:  Positive
Sentence:  I had a great day today.  The weather was beautiful and I spent time with friends.
Class label:  Negative
Sentence:  I was really disappointed by the latest superhero movie.  I would not recommend it.

Task:  Given a dialogue, classify whether the user is satisfied with the service.  You should
respond with "Satisfied" or "Unsatisfied".
Class label:  Satisfied
Dialogue:
- Agent:  Thank you for your feedback.  We will work to improve our service in the future.
- Customer:  I am happy with the service you provided.  Thank you for your help.
Class label:  Unsatisfied
Dialogue:
- Agent:  Sorry that we will cancel your order.  You will get a refund within 7 business days.
- Customer:  oh that takes too long.  I want you to take quicker action on this.

Task:  Which of the following is not an input type?  (a) number (b) date (c) phone number (d)
email address (e) all of these are valid inputs.
Class label:  (e)

Task:  {instruction for the target task}
```

Table 8: Prompt used for the output-first approach of instance generation. The model is prompted to generate the class label first, and then generate the corresponding input. This prompt is used for generating the instances for classification tasks.

# Results




| | Model | # Params | ROUGE-L |
|---|---|---|---|
| | **Vanilla LMs** | | |
| | T5-LM | 11B | 25.7 |
| | GPT3 | 175B | 6.8 |
| | **Instruction-tuned w/o SUPERNI** | | |
| ① | T0 | 11B | 33.1 |
| ① | GPT3 + T0 Training | 175B | 37.9 |
| ② | $GPT3_{SELF\text{-}INST}$ (Ours) | 175B | 39.9 |
| ② | $InstructGPT_{001}$ | 175B | **40.8** |
| | **Instruction-tuned w/ SUPERNI** | | |
| | T*k*-INSTRUCT | 11B | 46.0 |
| ③ | GPT3 + SUPERNI Training | 175B | 49.5 |
| ③ | $GPT3_{SELF\text{-}INST}$ + SUPERNI Training (Ours) | 175B | **51.6** |

Table 3: Evaluation results on *unseen* tasks from SUPERNI (§4.3). From the results, we see that ① SELF-INSTRUCT can boost GPT3 performance by a large margin (+33.1%) and ② nearly matches the performance of $InstructGPT_{001}$. Additionally, ③ it can further improve the performance even when a large amount of labeled instruction data is present.

# Alpaca: Results

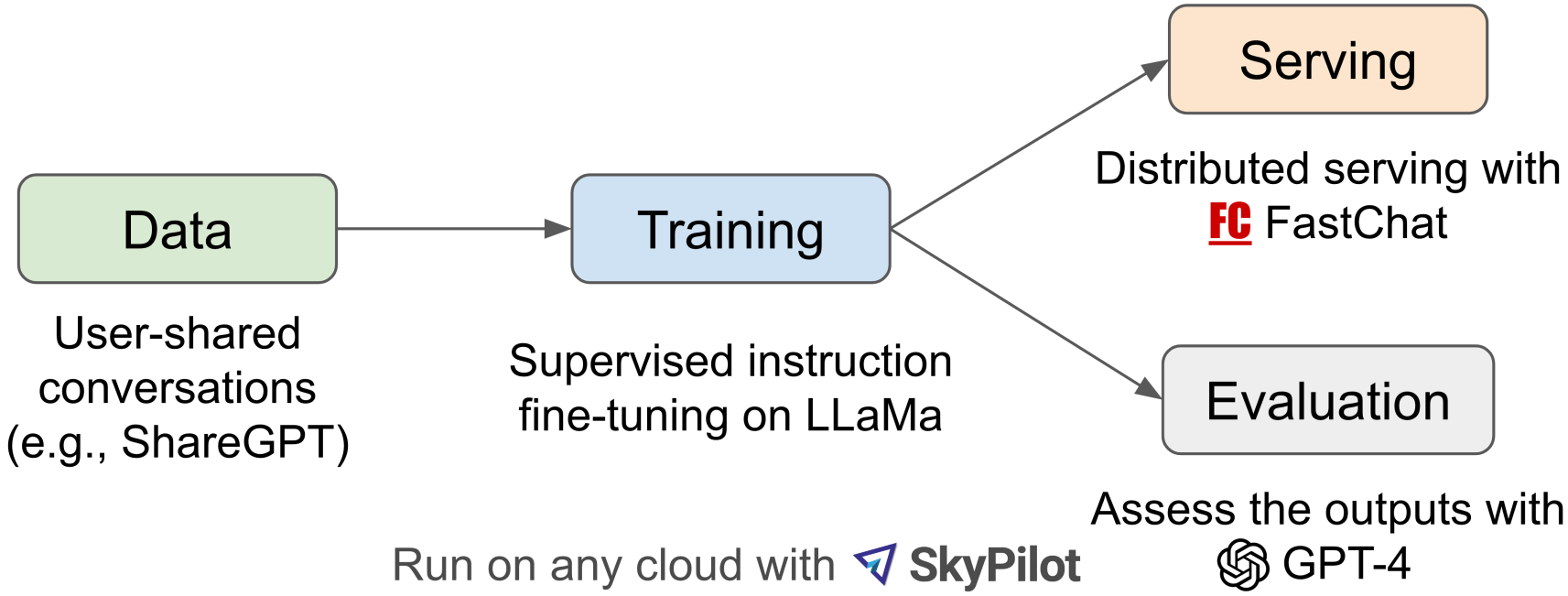
For our initial run, fine-tuning a 7B LLaMA model took 3 hours on 8 80GB A100s, which costs less than $100 on most cloud compute providers. We note that training efficiency can be improved to further reduce the cost.

We performed a blind pairwise comparison between **text-davinci-003 and Alpaca 7B**, and we found that these two models have very similar performance: Alpaca wins 90 versus 89 comparisons against text-davinci-003.

# Vicuna

Released Mar 30, 2023

We introduce Vicuna-13B, an open-source chatbot trained by fine-tuning LLaMA on **user-shared conversations collected from ShareGPT**. Preliminary evaluation using GPT-4 as a judge shows Vicuna-13B achieves more than 90%* quality of OpenAI ChatGPT and Google Bard while outperforming other models like LLaMA and Stanford Alpaca in more than 90%* of cases. The cost of training Vicuna-13B is around $300.



# Model Comparison

Table 1. Comparison between several notable models

| Model Name | LLaMA | Alpaca | Vicuna | Bard/ChatGPT |
|---|---|---|---|---|
| Dataset | Publicly available datasets (1T token) | Self-instruct from davinci-003 API (52K samples) | User-shared conversations (70K samples) | N/A |
| Training code | N/A | Available | Available | N/A |
| Evaluation metrics | Academic benchmark | Author evaluation | GPT-4 assessment | Mixed |
| Training cost (7B) | 82K GPU-hours | $500(data)+100$ (training) | $140 (training) | N/A |
| Training cost (13B) | 135K GPU-hours | N/A | $300 (training) | N/A |

# Model Performance

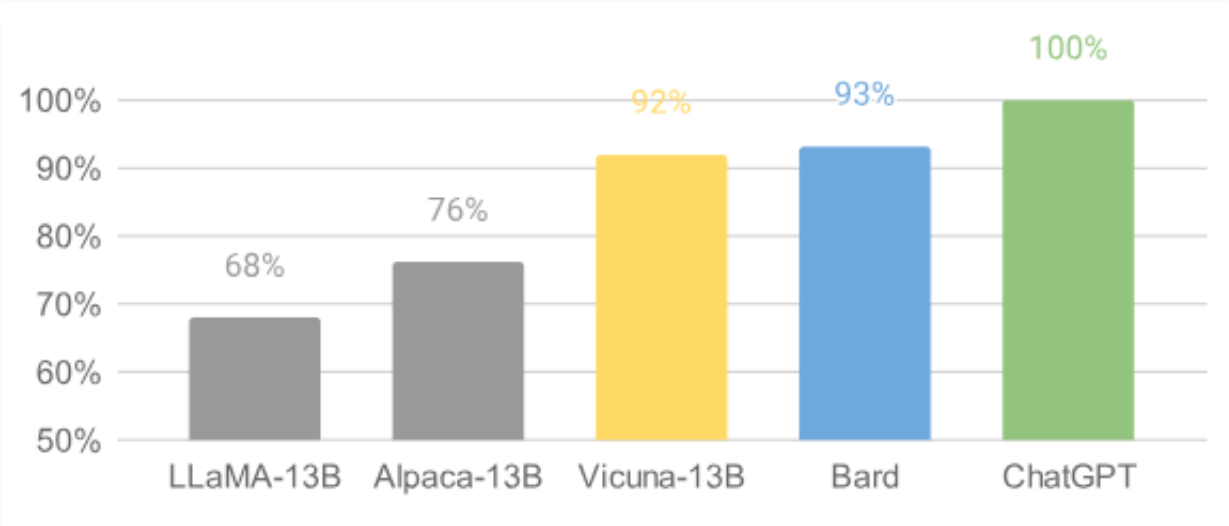



Figure 1. Relative Response Quality Assessed by GPT-4*

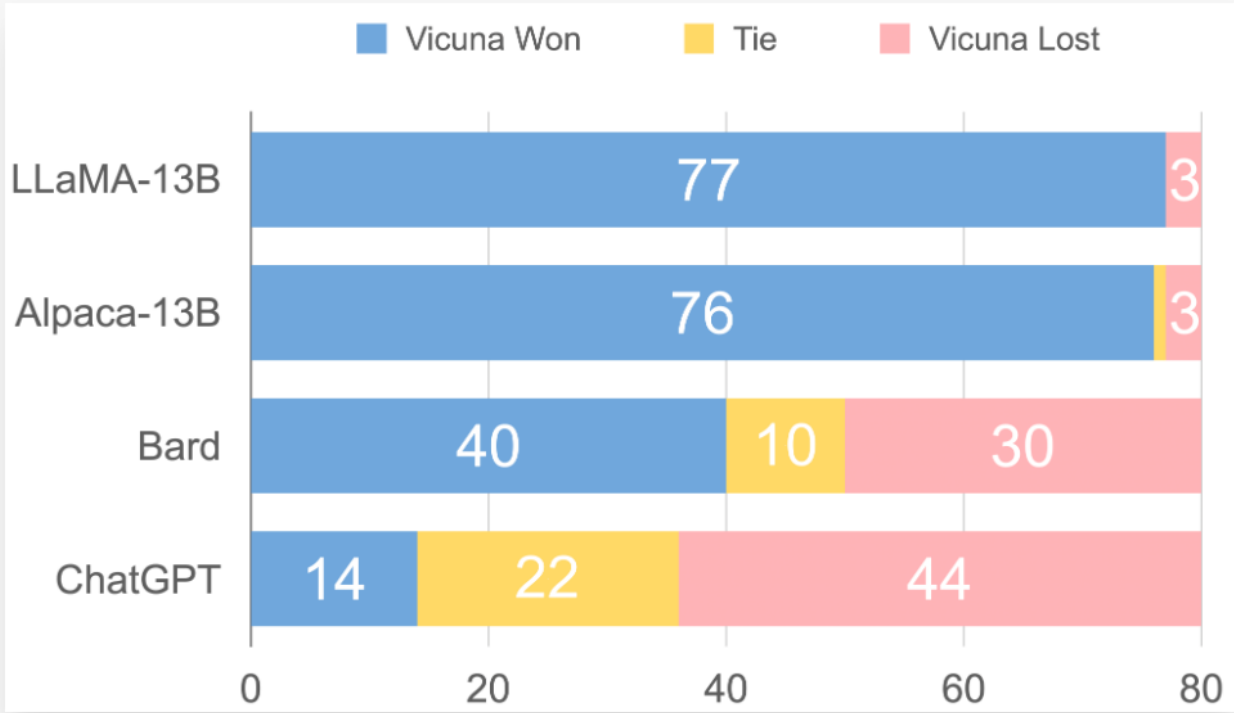



Figure 3. Response Comparison Assessed by GPT-4

While this proposed framework based on GPT-4 shows a potential to automate chatbot assessment, **it is not yet a rigorous approach**. Building an evaluation system for chatbots remains an open question requiring further research.

# Releases

Vicuna demo and Chatbot Arena website

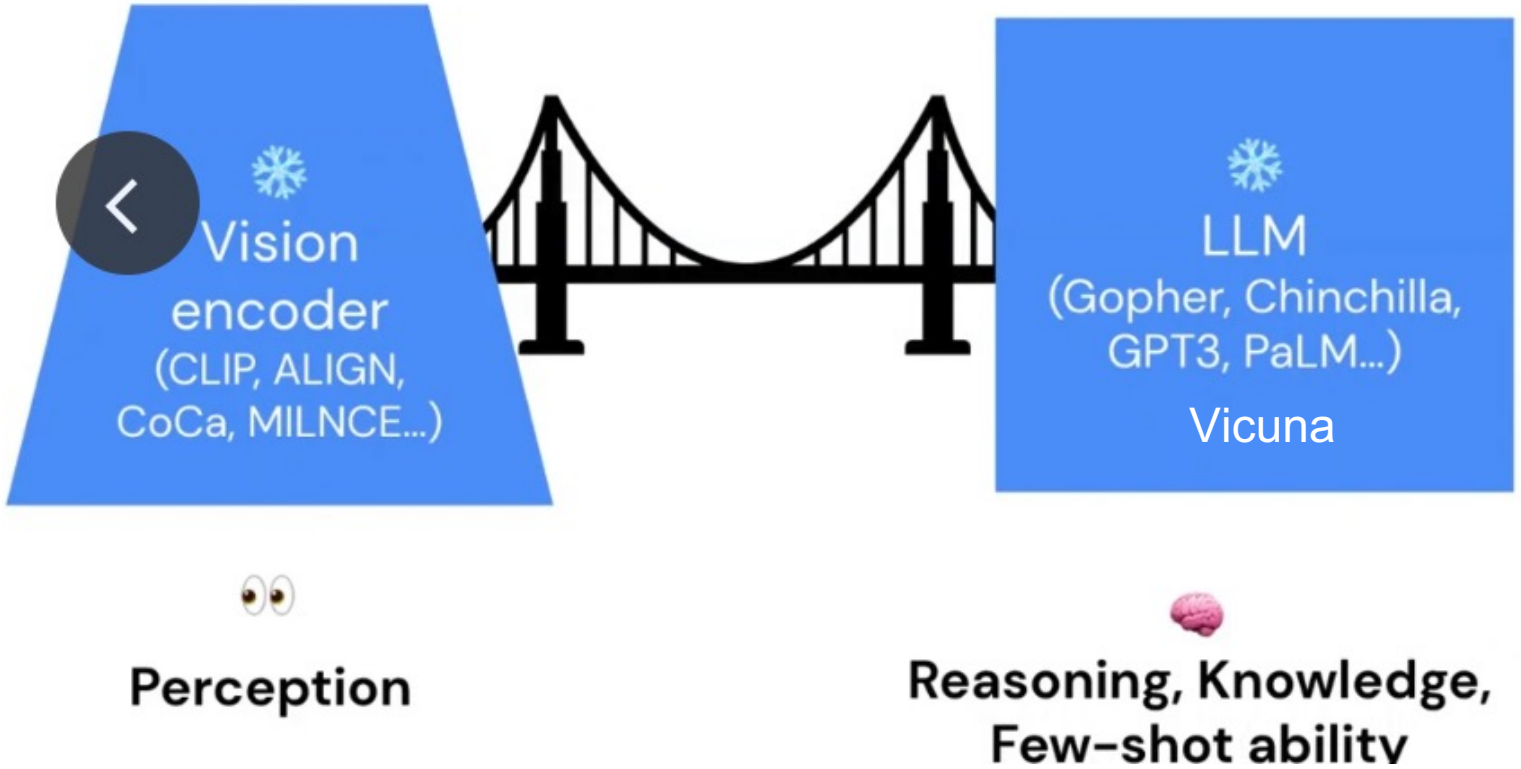
- [2024/03] 🔥 We released Chatbot Arena technical report.
- [2023/09] We released **LMSYS-Chat-1M**, a large-scale real-world LLM conversation dataset. Read the report.
- [2023/08] We released **Vicuna v1.5** based on Llama 2 with 4K and 16K context lengths. Download weights.
- [2023/07] We released **Chatbot Arena Conversations**, a dataset containing 33k conversations with human preferences. Download it here.

▼More

- [2023/08] We released **LongChat v1.5** based on Llama 2 with 32K context lengths. Download weights.
- [2023/06] We introduced **MT-bench**, a challenging multi-turn question set for evaluating chatbots. Check out the blog post.
- [2023/06] We introduced **LongChat**, our long-context chatbots and evaluation tools. Check out the blog post.
- [2023/05] We introduced **Chatbot Arena** for battles among LLMs. Check out the blog post.
- [2023/03] We released **Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90% ChatGPT Quality**. Check out the blog post.

# LLaVA: Large Language and Vision Assistant

Bridge the gap between the vision and LLM



# LLaVA: Large Language and Vision Assistant

Vicuna

pre-trained CLIP visual encoder ViT-L/14

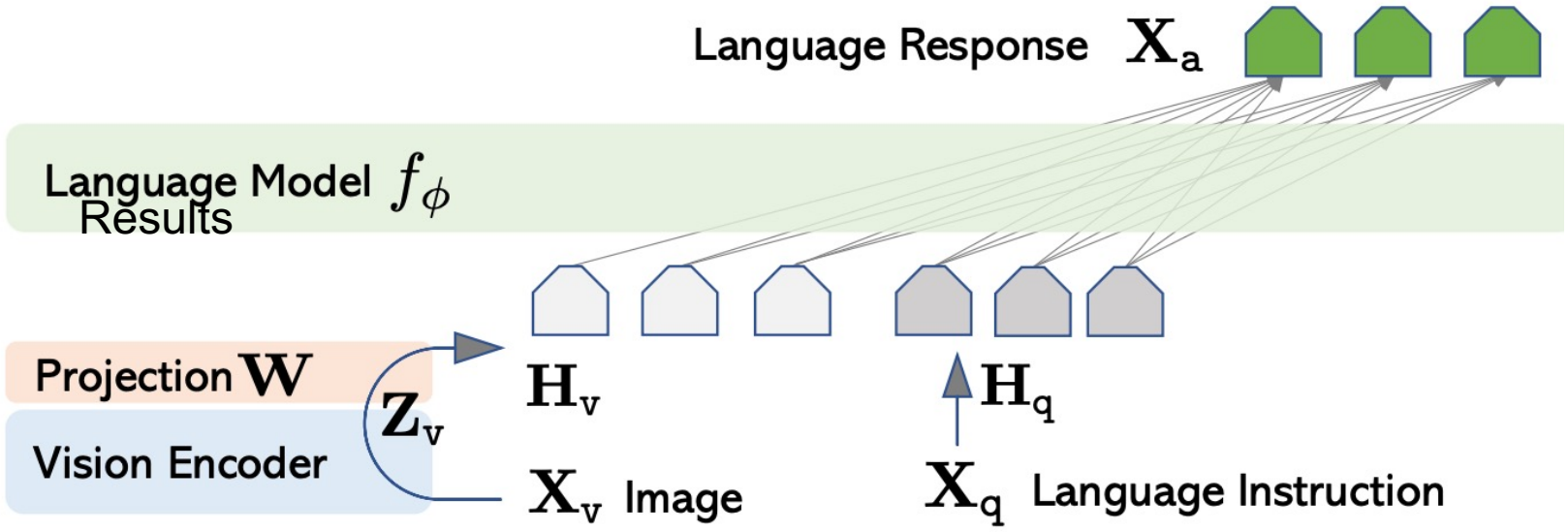



Figure 1: LLaVA network architecture.

# Pre-training for Feature Alignment.

Vicuna

pre-trained CLIP visual encoder ViT-L/14

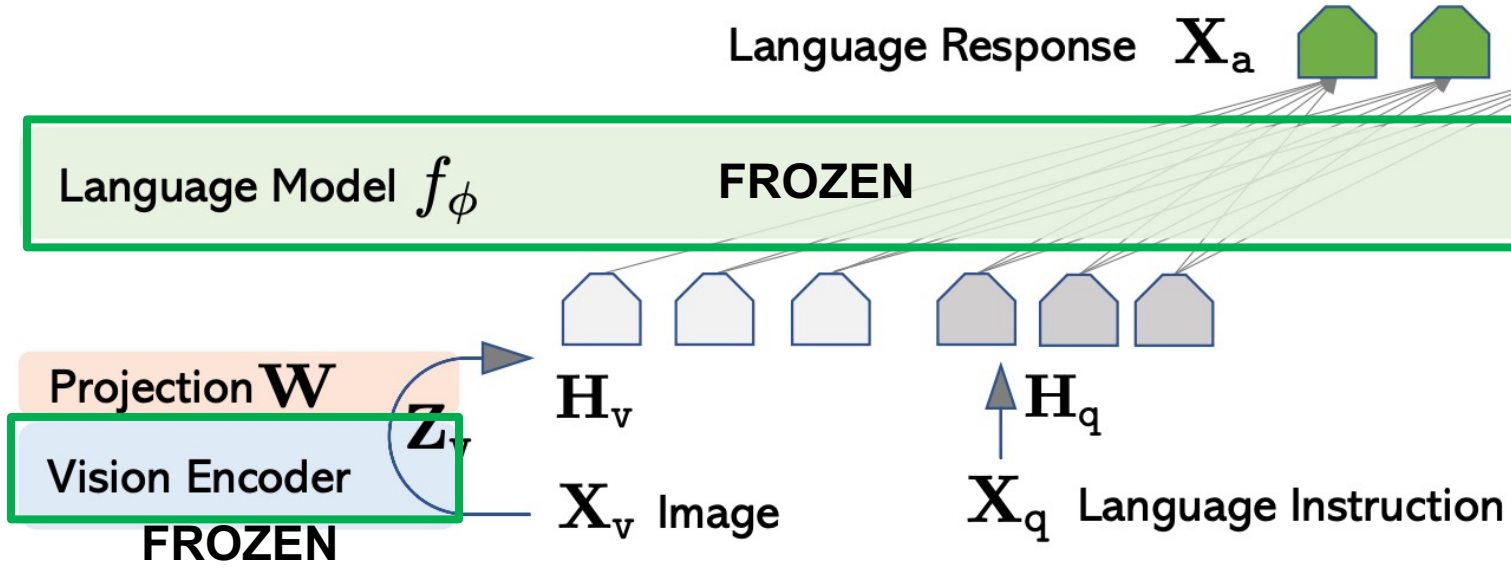



Figure 1: LLaVA network architecture.

Language model and Vision Encoder are frozen, only updating projection matrix W.

# Fine-tuning End-to-End

Vicuna

pre-trained CLIP visual encoder ViT-L/14

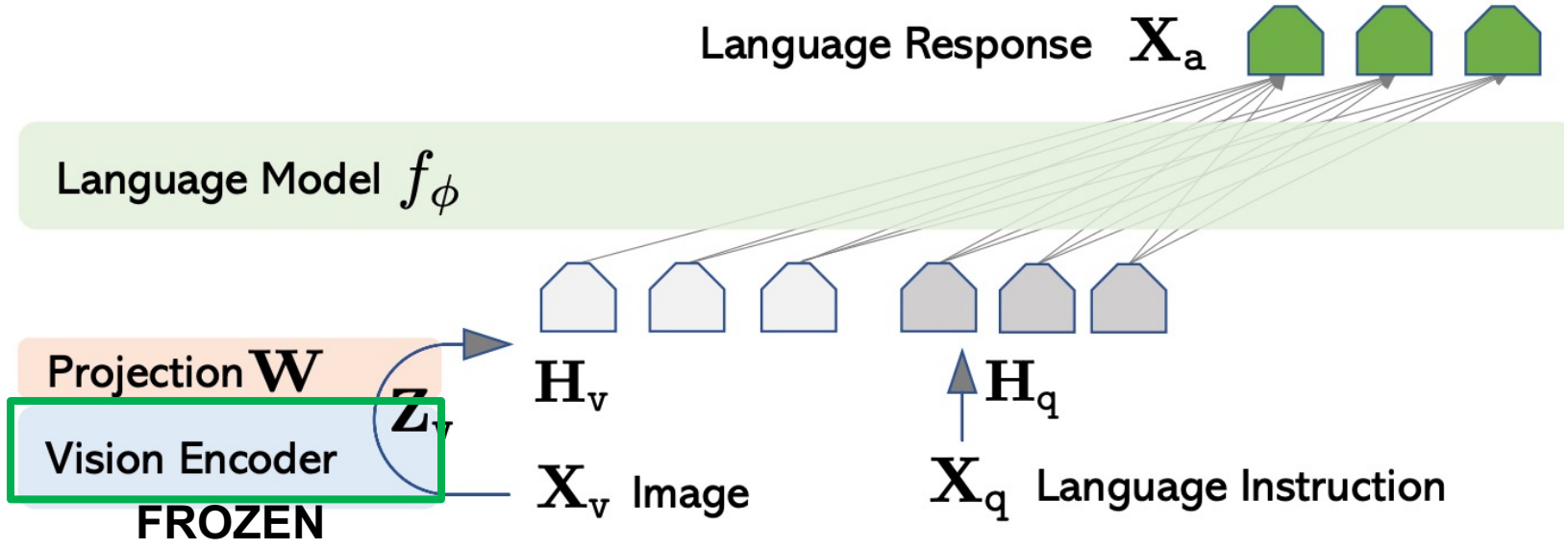
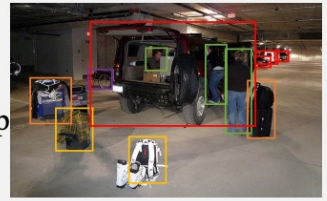



Figure 1: LLaVA network architecture.

Vision Encoder is frozen, continuing to update both the W and the Language Model in LLaVA

# Fine-tuning Datasets

**Context type 1: Captions**
A group of people standing outside of a black vehicle with various luggage.
Luggage surrounds a vehicle in an underground parking area
People try to fit all of their luggage in an SUV.
The sport utility vehicle is parked in the public garage, being packed for a trip
Some people with luggage near a van that is transporting it.



**Context type 2: Boxes**
person: [0.681, 0.242, 0.774, 0.694], backpack: [0.384, 0.696, 0.485, 0.914], suitcase: ...<omitted>

---

**Response type 1: conversation**
Question: What type of vehicle is featured in the image?
Answer: The image features a black sport utility vehicle (SUV) ...<omitted>
**Response type 2: detailed description**
The image is an underground parking area with a black sport utility vehicle (SUV) parked. There are three people in the scene, with one person standing closer to the left side of the vehicle, another person in the middle, and the third person on the right side. They are all working together to pack their luggage into the SUV for a trip. ...<omitted>
**Response type 3: complex reasoning**
Question: What challenges do these people face?
Answer: In the image, a group of people is standing outside a black SUV in a parking area, surrounded by various pieces of luggage, including suitcases and backpacks. They are facing the challenge of fitting all their luggage into the black SUV. There are multiple suitcases and backpacks to be packed, which suggests that the group has a significant amount of belongings ...<omitted>

Table 1: One example to illustrate the instruction-following data. The top block shows the contexts such as captions and boxes used to prompt GPT, and the bottom block shows the three types of responses. Note that the visual image is not used to prompt GPT, we only show it here as a reference.

Leverage ChatGPT/GPT-4 for multimodal instruction-following data collection:

(i) Input to GPT-4: Captions and Bounding boxes

(ii) prompt GPT-4 to curate a list of questions with the intent to instruct the assistant to describe the image content

(iii) We collect 158K unique language-image instruction-following samples in total, including 58K in conversations, 23K in detailed description, and 77k in complex reasoning respectively.

# The prompt to generate image-based conversation from ChatGPT/GPT-4

```
messages = [ {"role":"system", "content": f"""You are an AI visual assistant, and you are seeing a single image. What you see are provided with five sentences, describing the same image you are looking at. Answer all questions as you are seeing the image.

Design a conversation between you and a person asking about this photo. The answers should be in a tone that a visual AI assistant is seeing the image and answering the question. Ask diverse questions and give corresponding answers.

Include questions asking about the visual content of the image, including the object types, counting the objects, object actions, object locations, relative positions between objects, etc. Only include questions that have definite answers:
(1) one can see the content in the image that the question asks about and can answer confidently;
(2) one can determine confidently from the image that it is not in the image. Do not ask any question that cannot be answered confidently.

Also include complex questions that are relevant to the content in the image, for example, asking about background knowledge of the objects in the image, asking to discuss about events happening in the image, etc. Again, do not ask about uncertain details. Provide detailed answers when answering complex questions. For example, give detailed examples or reasoning steps to make the content more convincing and well-organized. You can include multiple paragraphs if necessary."""}
]
for sample in fewshot_samples:
    messages.append({"role":"user", "content":sample['context']})
    messages.append({"role":"assistant", "content":sample['response']} )
messages.append({"role":"user", "content":'\n'.join(query)})
```

# Results

**Visual input example, Extreme Ironing:**

Source: https://www.barnorama.com/wp-content/uploads/2016/12/03-Confusing-Pictures.jpg

| | |
|---|---|
| User | What is unusual about this image? |
| LLaVA | The unusual aspect of this image is a man ironing clothes on the back of a minivan or van. This is not a typical place to perform this activity, as one would usually iron clothes in a more stationary and safe location, such as a home, using a regular ironing board. The scene depicted in the image is peculiar as it involves a makeshift ironing setup on a vehicle, which can be both unsafe and unconventional. Additionally, it is not clear how the man is able to maintain balance and stability while ironing clothes in such an unstable environment. |
| | [*Start a new conversation, and clear the history*] |
| User | What's happening in the scene? |
| LLaVA | The image depicts a man standing on top of a yellow SUV in a busy city street. He is holding a portable ladder, seemingly ironing clothes while standing on the vehicle. Around the scene, there are other cars, a traffic light, and a fire hydrant. Additionally, there are two other people in the vicinity, one close to the man on the car and another further away. The man on the ladder appears to be attracting attention due to his unconventional choice of ironing his clothes on top of a moving car. The city street around him is bustling with activity, adding to the unique nature of the scene. |
| User | What is unusual about this image? |
| GPT-4 [36] | The unusual thing about this image is that a man is ironing clothes on an ironing board attached to the roof of a moving taxi. |

# Results: LLaVA-Bench

GPT-4 as the evaluator

| | Conversation | Detail description | Complex reasoning | **All** |
|---|---|---|---|---|
| Full data | 83.1 | 75.3 | 96.5 | 85.1 |
| Detail + Complex | 81.5 (-1.6) | 73.3 (-2.0) | 90.8 (-5.7) | 81.9 (-3.2) |
| Conv + 5% Detail + 10% Complex | 81.0 (-2.1) | 68.4 (-7.1) | 91.5 (-5.0) | 80.5 (-4.4) |
| Conversation | 76.5 (-6.6) | 59.8 (-16.2) | 84.9 (-12.4) | 73.8 (-11.3) |
| No Instruction Tuning | 22.0 (-61.1) | 24.0 (-51.3) | 18.5 (-78.0) | 21.5 (-63.6) |

Table 4: Ablation on LLaVA-Bench (COCO) with different training data. We report relative scores *w.r.t.* a text-only GPT-4 model that uses ground truth image captions and bounding boxes as visual input. We prompt GPT-4 with the answers from our model outputs and the answers by GPT-4 (text-only), and let it compare between both responses and give a rating with an explanation.

| | Conversation | Detail description | Complex reasoning | All |
|---|---|---|---|---|
| OpenFlamingo [5] | $19.3 \pm 0.5$ | $19.0 \pm 0.5$ | $19.1 \pm 0.7$ | $19.1 \pm 0.4$ |
| BLIP-2 [28] | $54.6 \pm 1.4$ | $29.1 \pm 1.2$ | $32.9 \pm 0.7$ | $38.1 \pm 1.0$ |
| LLaVA | $57.3 \pm 1.9$ | $52.5 \pm 6.3$ | $81.7 \pm 1.8$ | $67.3 \pm 2.0$ |
| LLaVA$^\dagger$ | $58.8 \pm 0.6$ | $49.2 \pm 0.8$ | $81.4 \pm 0.3$ | $66.7 \pm 0.3$ |

Table 5: Instruction-following capability comparison using relative scores on LLaVA-Bench (In-the-Wild). The results are reported in the format of *mean* $\pm$ *std*. For the first three rows, we report three inference runs. LLaVA performs significantly better than others. $^\dagger$ For a given set of LLaVA decoding sequences, we evaluate by querying GPT-4 three times; GPT-4 gives a consistent evaluation.

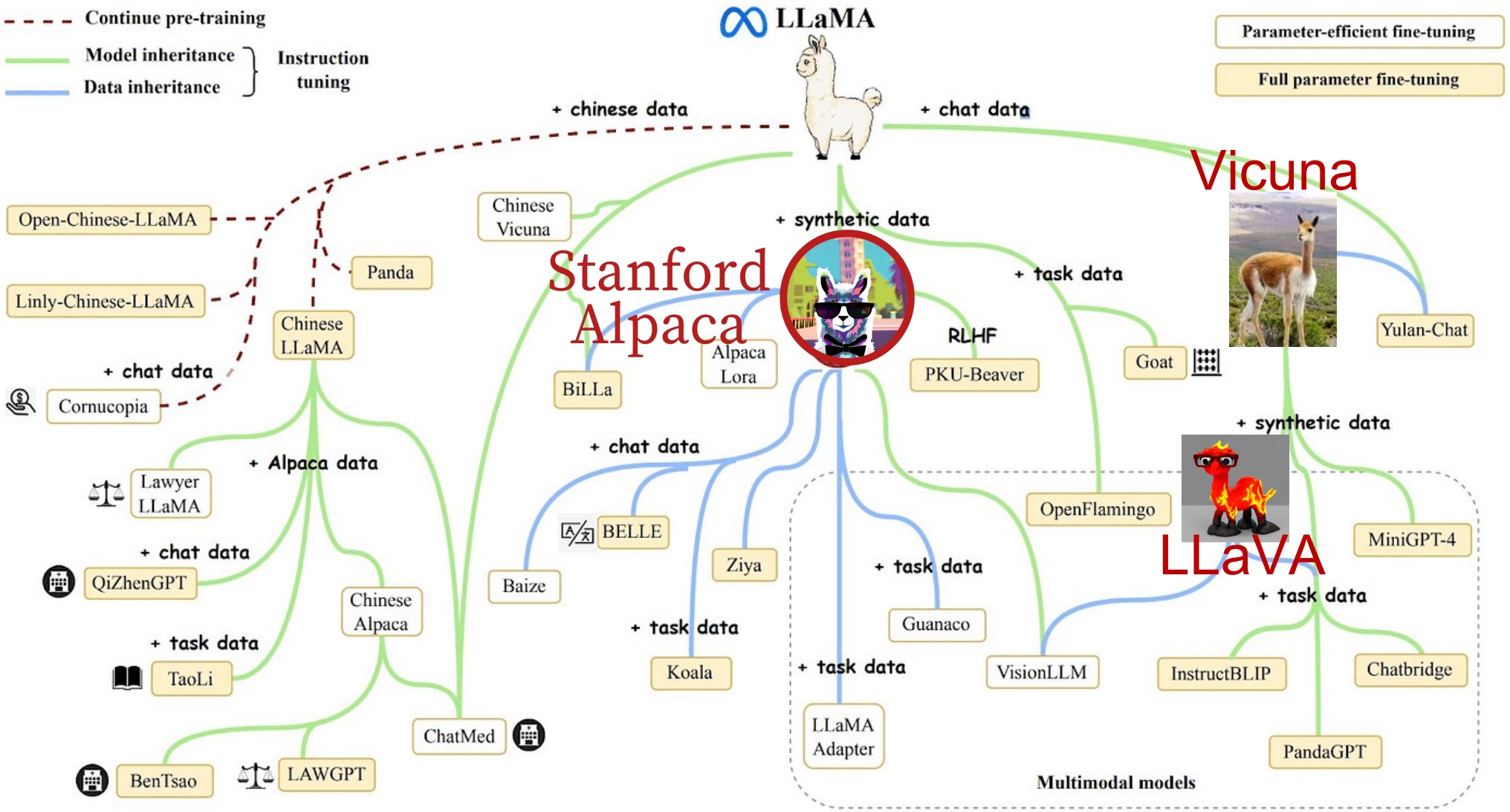
# Results: Science QA

CoT: Chain of Thought

| Method | Subject | | | Context Modality | | | Grade | | Average |
|---|---|---|---|---|---|---|---|---|---|
| | NAT | SOC | LAN | TXT | IMG | NO | G1-6 | G7-12 | |
| *Representative & SoTA methods with numbers reported in the literature* | | | | | | | | | |
| Human [34] | 90.23 | 84.97 | 87.48 | 89.60 | 87.50 | 88.10 | 91.59 | 82.42 | 88.40 |
| GPT-3.5 [34] | 74.64 | 69.74 | 76.00 | 74.44 | 67.28 | 77.42 | 76.80 | 68.89 | 73.97 |
| GPT-3.5 w/ CoT [34] | 75.44 | 70.87 | 78.09 | 74.68 | 67.43 | 79.93 | 78.23 | 69.68 | 75.17 |
| LLaMA-Adapter [59] | 84.37 | 88.30 | 84.36 | 83.72 | 80.32 | 86.90 | 85.83 | 84.05 | 85.19 |
| MM-CoT$_{Base}$ [61] | 87.52 | 77.17 | 85.82 | 87.88 | 82.90 | 86.83 | 84.65 | 85.37 | 84.91 |
| MM-CoT$_{Large}$ [61] | 95.91 | 82.00 | 90.82 | 95.26 | 88.80 | 92.89 | 92.44 | 90.31 | 91.68 |
| *Results with our own experiment runs* | | | | | | | | | |
| GPT-4$^{\dagger}$ | 84.06 | 73.45 | 87.36 | 81.87 | 70.75 | 90.73 | 84.69 | 79.10 | 82.69 |
| LLaVA | 90.36 | 95.95 | 88.00 | 89.49 | 88.00 | 90.66 | 90.93 | 90.90 | 90.92 |
| LLaVA+GPT-4$^{\dagger}$ (complement) | 90.36 | 95.50 | 88.55 | 89.05 | 87.80 | 91.08 | 92.22 | 88.73 | 90.97 |
| LLaVA+GPT-4$^{\dagger}$ (judge) | 91.56 | 96.74 | 91.09 | 90.62 | 88.99 | 93.52 | 92.73 | 92.16 | **92.53** |

Table 7: Accuracy (%) on Science QA dataset. Question categories: NAT = natural science, SOC = social science, LAN = language science, TXT = text context, IMG = image context, NO = no context, G1-6 = grades 1-6, G7-12 = grades 7-12. $^{\dagger}$Text-only GPT-4, our eval. Our novel model ensembling with the text-only GPT-4 consistently improves the model's performance under all categories, setting the new SoTA performance.

# LLaMA Family




Ref: https://blackbearlabs.ai/blog-detail/open-large-language-models-history-2023-report

# LLaMA series

- LLaMA: Open and Efficient Foundation Language Models, Feb 2023
- LLaMA 2: Open Foundation and Fine-Tuned Chat Models, July 2023
- Variants of LLaMA: Alpaca, Vicuna, LLaVA
- Hands-on session: Fine-tune a LLaMA model