What is Llama?

WORKING WITH LLAMA 3

Imtihan Ahmed
Machine Learning Engineer

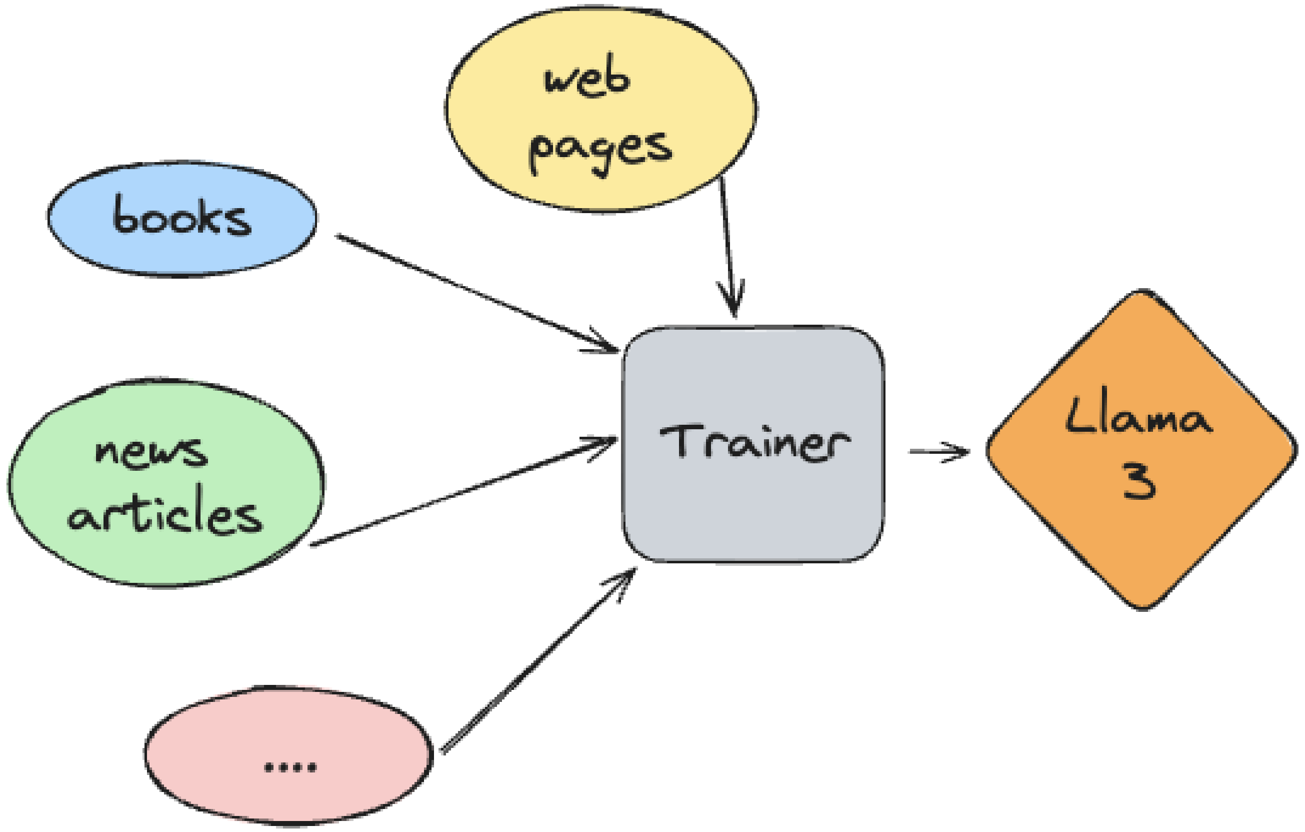
datacamp

What is Llama 3 under the hood

- Open-source large language model (LLM)
- Built by Meta
- 128,256 token vocabulary
- 8,192 token context
- At least 4 variants
  - 7B + 7B instruct
  - 70B + 70B instruct

How was Llama 3 Trained

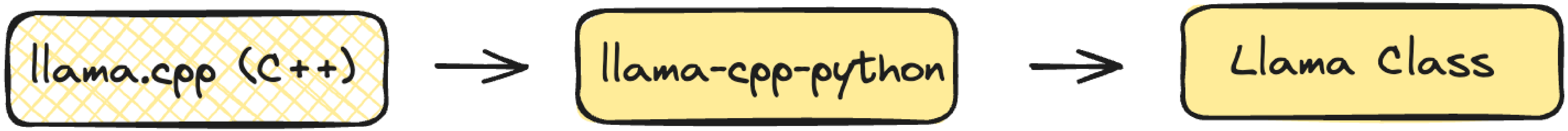
- Trained on 15 trillion tokens
- 30 languages
- 24,000 GPU



- Supervised fine-tuning (SFT)
- Rejection Sampling
- Proximal Policy Optimization (PPO)
- Direct Preference Optimization (DPO)

Using Llama 3

- `llama-cpp-python`
- Python wrapper for `llama.cpp`
- `Llama` class
  - access llama parameters
  - interface to `llama.cpp` code



Running Llama 3 with llama_cpp

```python
from llama_cpp import Llama
path_to_model="path/to/model.gguf"
llm = Llama(model_path=path_to_model)
output = llm("Where do llamas live?",)
print(output)
```

```
{'id': 'cmpl-af88304f-97b0-49f5-ba20-db87f86c4068',
 'object': 'text_completion',
 'created': 1715222298,
 'model': './Llama3-gguf-unsloth.Q4_K_M.gguf',
 'choices': [{'text': ' Llamas are domesticated animals and can be found in every continent,
     ...
}
```

Extracting model outputs

```python
print(output)
```

```
{'id': 'cmpl-af88304f-97b0-49f5-ba20-db87f86c4068',
 'object': 'text_completion',
 'created': 1715222298,
 'model': './Llama3-gguf-unsloth.Q4_K_M.gguf',
 'choices': [{'text': ' Llamas are domesticated animals and can be found in every continent,
     ...
}
```

```python
output_text = output['choices'][0]['text']
```

Asking Llama 3 a question and controlling its result

```python
from llama_cpp import Llama
llm = Llama(model_path="./Llama3-gguf-unsloth.Q4_K_M.gguf", n_gpu_layers=-1)
output = llm(
      "Q: Name 5 species of llamas? A: ",
      max_tokens=32,
      stop=["Q:", "\n"],
)
print(output['choices'][0]['text'].split('A: ')[1])
```

```
'1) Guanaco, 2) Bactrian Camel, 3) Alpaca, 4) Llama, and 5) Vic'
```

Let's practice!

WORKING WITH LLAMA 3

Getting started with Llama

WORKING WITH LLAMA 3

Imtihan Ahmed
Machine Learning Engineer

How to use Llama

- Python bindings for `llama.cpp`
- Python 3.8+
- C compiler (gcc/clang, Visual Studio, Xcode)
- `pip install llama-cpp-python`

- supported models:
  - LLaMA
  - LLaMA 2
  - LLaMA 3
  - Mistral
  - Falcon
  - and many more..

[1] https://github.com/abetlen/llama-cpp-python

Loading models locally with llama_cpp

```
from llama_cpp import Llama
llm = Llama(
      model_path="./models/7B/llama-model.gguf",
      n_gpu_layers=-1, # load to GPU
      seed=1337, # set random seed
      n_ctx=2048, # set context size
)
```

Loading models from Hugging Face

- `pip install huggingface-hub`

```python
from llama_cpp import Llama
llm = Llama.from_pretrained(
    repo_id="Qwen/Qwen1.5-0.5B-Chat-GGUF",
    filename="*q8_0.gguf"
)
```

Controlling how a model creates completions

```
output = llm(
        "Q: What is the circumference of the Earth? A:", # Prompt
        max_tokens=32,
        stop=["Q:", "\n"],
        temperature=0.9,
        repeat_penalty=1.3
)
```

```
history = [    # Instruct the model to behave like Plato
   {"role": "system", "content": "You are the Greek
philosopher Plato. Answer every question using his voice.
"},    # Identify that the following text is from the
user    {           "role": "user",           "content"
: "Can any shape that exist in the real world be perfect
and why?"     }]# Pass in conversation context to the
completion callresult = llm.create_chat_completion
(messages=history, max_tokens=50)print(result)
```

```
output = llm.create_completion("hello", max_tokens=32,...)
```

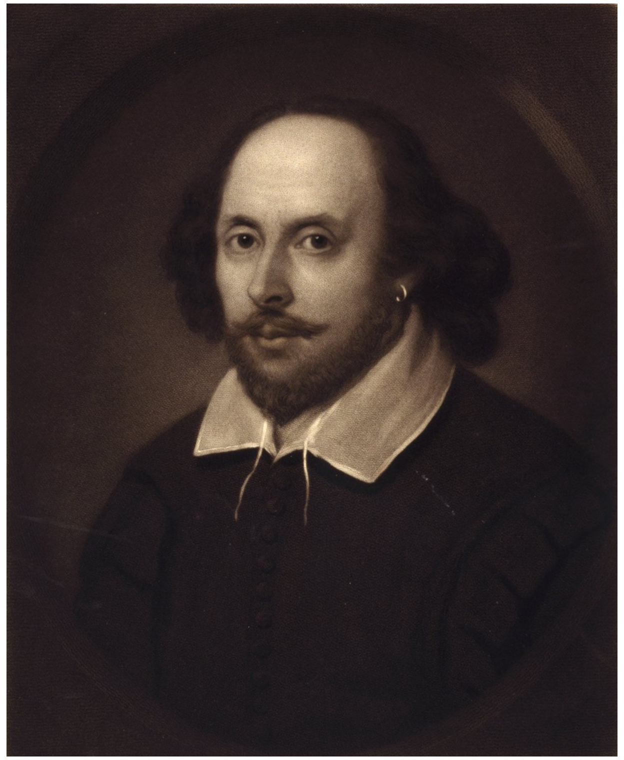
Creating chat completions

```
output = llm.create_chat_completion(
  messages = [
          {
            "role": "system",
            "content": "You are an assistant who speaks only Shakespearean"
          },
          {
            "role": "user",
            "content": "Describe New York in 10 words"
          }
        ]
    )
```

Chat completion result

```
print(chat_comp['choices'][0]['message']['content'])
```

```
'"Fair Gotham\'s bustling streets, a tapestry of urban delight."'
```



Let's practice!

WORKING WITH LLAMA 3

Prompt engineering Llama 3

WORKING WITH LLAMA 3

Imtihan Ahmed
Machine Learning Engineer

Is this a good prompt?

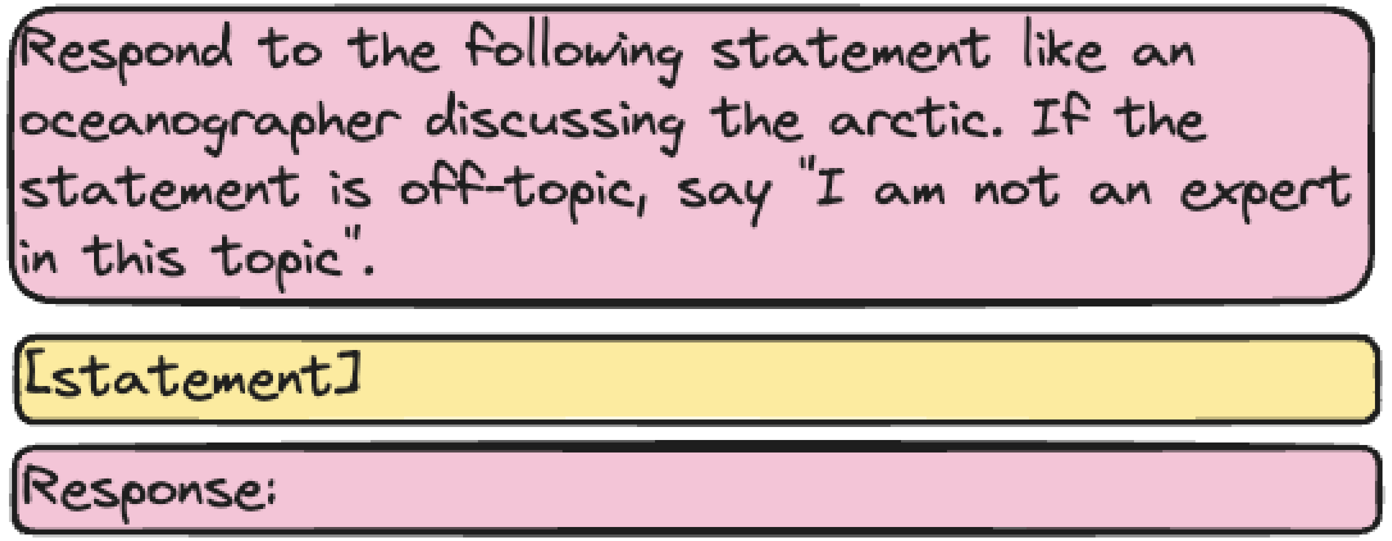
```
"Tell me about London."
```

```
"""List 5 tourist locations in London, UK and the times of the
year when they are popular and why.
Use bullet-points in your answer. """
```

Writing a good prompt

- Precise
- Short
- Direct
- Beginning or end
- Separated from the input text
- Completion keywords
- Chain-of-thought

Zero-shot learning



Respond to the following statement like an oceanographer discussing the arctic. If the statement is off-topic, say "I am not an expert in this topic".

[statement]

Response:

Few-shot prompting

- Prompt with examples
- Structured inputs and outputs
- Complex instructions

```
"""
Text: x is two fourty-two point five to the power of five
equation: x = 242.5^5
Text: x is nine thousand ninety three divided by three
equation:
"""
```

Writing a good summarization prompt

```
text = """Llamas are social animals and live with others as a herd.
Their wool is soft and contains only a small amount of lanolin.
Llamas can learn simple tasks after a few repetitions.
When using a pack, they can carry about 25 to 30% of their
body weight for 8 to 13 km (5 to 8 miles).
The name llama (in the past also spelled "lama" or "glama")
was adopted by European settlers from native Peruvians."""
prompt = f"""Instruction: Create a summary using less than 20 words of the following text.
Text: {text}
Summary:
"""
```

[1] https://en.wikipedia.org/wiki/Llama

Generating summaries with Llama

```
output = llm(
      prompt,
      max_tokens=32,
      stop=["Q:", "\n"],
)
print(output['choices'][0]['text'])
```

```
Llamas are social animals with soft wool, able to learn tasks
and carry loads up to 30% of their body weight.
```

Translations with few-shot prompting

```
text="""EN: Hello
FR: Bonjour
EN: Goodbye
FR: Au revoir
EN: Good day
FR:
"""
```

```
# Fill in the 3-
shot prompt (you
can use multiple
lines)     ="""
Review 1:
happyReview 2:
unhappyReview 4:
Delicious food,
and excellent
customer service!
Sentiment 4:"""
        =    (     ,
            =2,
=["Q:"]) print
(      ['choices']
[0]['text'])
```

```
output = llm(text, max_tokens=32, stop=["Q:", "\n"],)
print(output['choices'][0]['text'])
```

```
Bonne journée
```

Let's practice!

WORKING WITH LLAMA 3