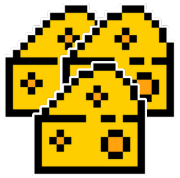
LLM Reasoning

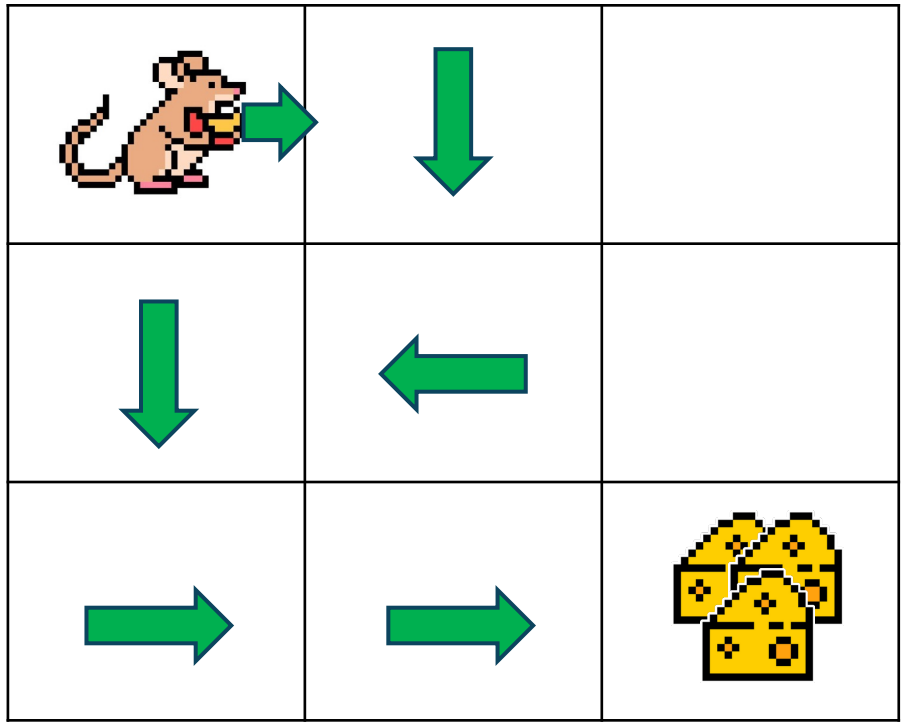
❖ Value-based methods

	-3	-10
-3	-2	-1
-2	-1	

Value-based methods: train the agent to learn which state is more valuable and take the action that leads to it.

LLM Reasoning

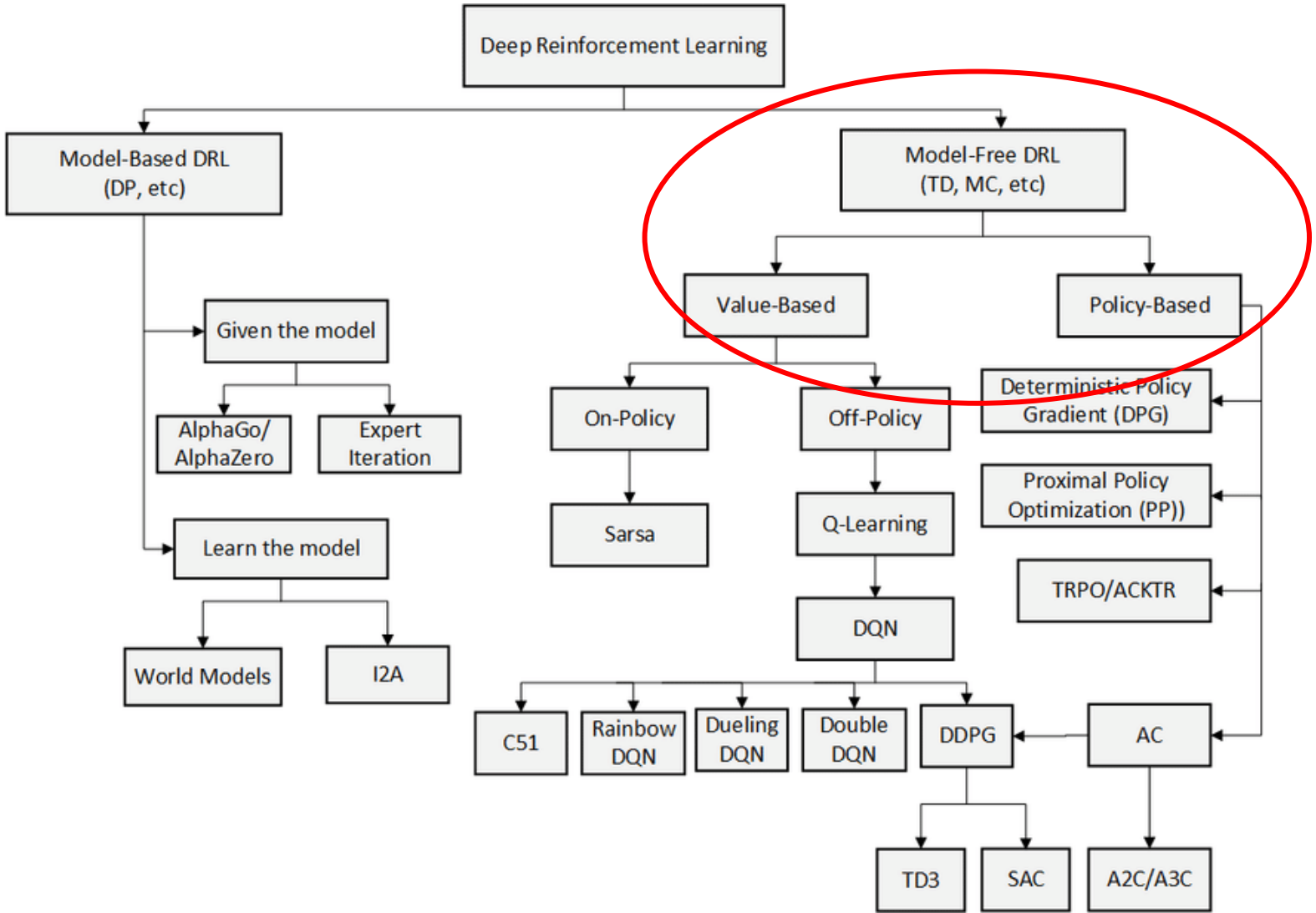
❖ Policy-based methods



Policy-based methods: train the agent to learn which action to take, given a state.

LLM Reasoning

❖ RL Algorithms Taxonomy



LLM Reasoning

❖ Introduction

Cha của Reggie đã cho anh ấy 48 đô la. Reggie đã mua 5 cuốn sách, mỗi cuốn có giá x. Reggie còn lại 38 tiền. Giá trị của biến x chưa biết là bao nhiêu?

Modify Prompt

Training Prompt

System

You are given a problem. Think about the problem and provide your thought process. Place it between <thinking> and </thinking>. Then, provide your final answer between <answer> and </answer>.

Question

Cha của Reggie đã cho anh ấy 48 đô la. Reggie đã mua 5 cuốn sách, mỗi cuốn có giá x. Reggie còn lại 38 tiền. Giá trị của biến x chưa biết là bao nhiêu?

Answer

2

Thinking ...

Step 1

Step 2

⋮

Step n

Training

Output

Thought

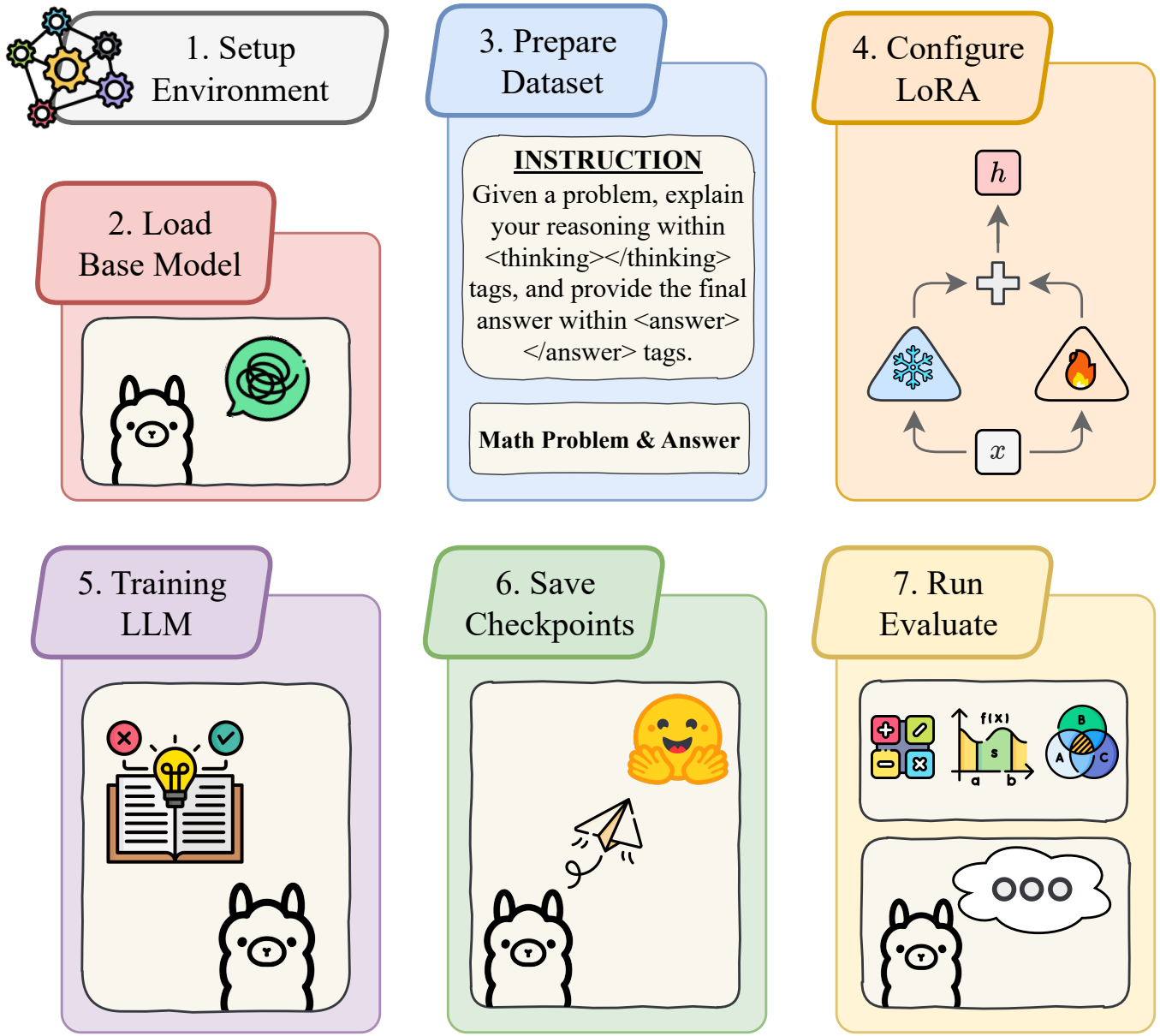
Reggie đã chi $48 - 38 = 10$ đô la cho 5 cuốn sách, nên lập phương trình $5x = 10$. Giải ra được $x = 2$. Vậy mỗi cuốn sách giá 2 đô la.

Answer

2

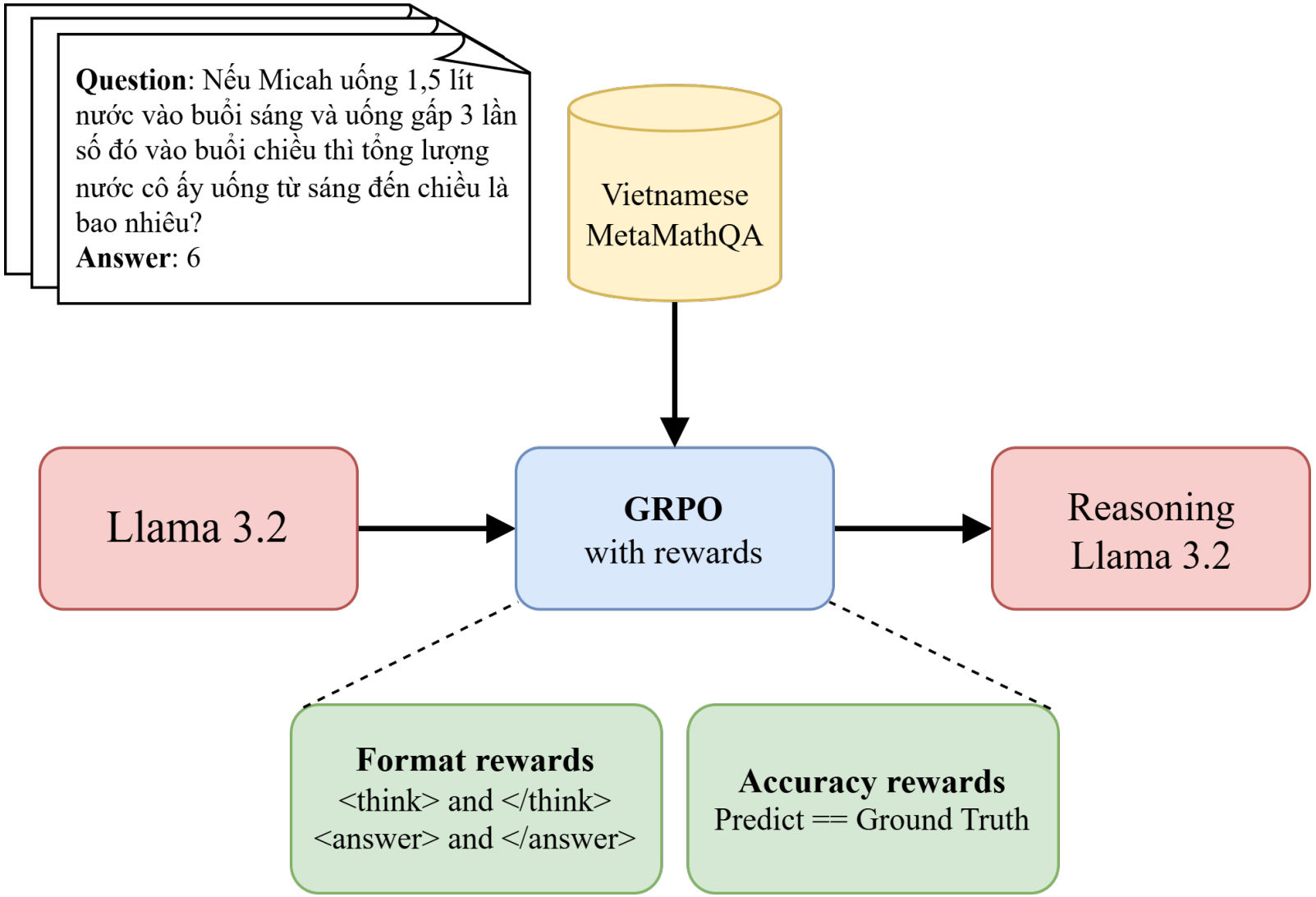
LLM Reasoning

❖ Pipeline



LLM Reasoning

❖ Training Math Reasoning



LLM Reasoning

❖ Step 1: Install and import necessary libraries

Unsloth is an open-source Python library that hand-writes GPU kernels and patches core ML frameworks to fine-tune large language models up to 2× faster while cutting GPU memory use by 70–80%.

vLLM is a high-throughput, memory-efficient LLM inference and serving engine from UC Berkeley, leveraging PagedAttention, continuous batching, speculative decoding, and multi-precision quantization support.

LLM Reasoning

❖ Step 1: Install and import necessary libraries

```
%pip install unsloth vllm==0.7.3
```

```
import re
from vllm import SamplingParams
from unsloth import FastLanguageModel
from datasets import load_dataset, Dataset
from trl import GRPOConfig, GRPOTrainer
```

LLM Reasoning

❖ Step 2: Load base model

```python
max_seq_length = 2048
lora_rank = 64

model, tokenizer = FastLanguageModel.from_pretrained(
    model_name="meta-llama/Llama-3.2-3B-Instruct",
    max_seq_length=max_seq_length,
    load_in_4bit=False,
    fast_inference=True,
    max_lora_rank=lora_rank,
    gpu_memory_utilization=0.8,
)
```

```python
model = FastLanguageModel.get_peft_model(
    model,
    r=lora_rank,
    target_modules=[
        "q_proj", "v_proj"
    ],
    lora_alpha=lora_rank,
    use_gradient_checkpointing="unsloth",
    random_state=3407,
)
```

https://llama-2.ai/llama-2-model-details/

LLM Reasoning

❖ Step 3: Load & Preprocess Dataset

```
dataset = load_dataset("5CD-AI/Vietnamese-meta-math-MetaMathQA-40K-gg-translated", split="train")
```

README.md: 100% 118/118 [00:00<00:00, 5.88kB/s]

MetaMathQA-40K_vi.json: 100% 69.2M/69.2M [00:00<00:00, 91.2MB/s]

Generating train split: 100% 40000/40000 [00:01<00:00, 31852.24 examples/s]

```
print("Dataset structure:", dataset)
```

```
Dataset structure: Dataset({
    features: ['response_vi', 'query_vi', 'response_en', 'type', 'query_en'],
    num_rows: 40000
})
```

LLM Reasoning

❖ Step 3: Load & Preprocess Dataset

Answers Questions

response_vi string · lengths 14 3.86k	query_vi string · lengths 15 2.31k	response_en string · lengths 15 3.74k	type string · classes 8 values	query_en string · lengths 16 2.28k
Để giải quyết vấn đề này, chúng ta cần xác định giá…	Cha của Reggie đã cho anh ấy 48 đô la. Reggie đã mu…	To solve this problem, we need to determine the…	GSM_SV	Reggie's father gave him $48. Reggie bought 5…
Có 28 quân domino trong bộ này. Jean và ba người bạn…	Jean và ba người bạn của cô ấy đang chơi trò chơi…	There are 28 dominoes in the set. Jean and her…	GSM_AnsAug	Jean and her three friends are playing a game of…
Cally có tổng cộng 10 áo trắng + 5 áo màu + 7 quần…	Cally và Danny giặt quần áo. Cally có 10 áo sơ mi…	Cally has a total of 10 white shirts + 5 colored…	GSM_AnsAug	Cally and Danny washed their clothes. Cally has…
Ban đầu Karen gọi một chiếc burger giá 5 đô la.…	Tổng chi phí cho đơn hàng đồ ăn nhanh của Karen là…	Karen initially ordered a 5-dollar burger. She then…	GSM_Rephrased	What is the total cost of Karen's fast-food order i…
Để xác định giá trị của số thứ 40 trong cách sắp xếp…	Các số 20 đầu tiên của sự sắp xếp được hiển thị…	To determine the value of the 40th number in the…	MATH_SV	The first 20 numbers of an arrangement are shown…
Simon hái tổng cộng 100 + 200 = 300 quả việt quất.…	Nếu Simon hái 100 quả việt quất từ bụi cây của mình…	Simon picks a total of 100 + 200 = 300 blueberries.…	GSM_Rephrased	If Simon picks 100 blueberries from his own…

‹ Previous 1 2 3 ... 400 Next ›

Vietnamese-meta-math-MetaMathQA-40K-gg-translated Dataset

LLM Reasoning

❖ Chat-style model: Conversation

```
[
  {
    "role": "system",
    "content": "You are a helpful assistant that summarizes content clearly."
  },
  {
    "role": "user",
    "content": "Please summarize the following:\n\nMachine learning is a field of AI that allows computers to learn from data without being explicitly programmed."
  },
  {
    "role": "assistant",
    "content": "Machine learning helps computers learn from data automatically, without needing explicit instructions."
  }
]
```

Single-turn

```
[
  {
    "role": "system",
    "content": "You are a helpful assistant that summarizes content clearly."
  },
  {
    "role": "user",
    "content": "Please summarize the following:\n\nMachine learning is a field of AI that allows computers to learn from data without being explicitly programmed."
  },
  {
    "role": "assistant",
    "content": "Machine learning helps computers learn from data automatically, without needing explicit instructions."
  },
  {
    "role": "user",
    "content": "Can you also summarize this?\n\nDeep learning is a subset of machine learning that uses neural networks with many layers."
  },
  {
    "role": "assistant",
    "content": "Deep learning is a type of machine learning that uses multi-layered neural networks to learn complex patterns from data."
  }
]
```

Multi-turn

LLM Reasoning

❖ Llama 3.2 Prompt Template

Supported Roles: There are 4 different roles that are supported by Llama text models: system, assistant, user, ipython.

[system, assistant, user, ipython]

Supported Role	Description
system	Sets the context in which to interact with the AI model. It typically includes rules, guidelines, or necessary information that help the model respond effectively.
user	Represents the human interacting with the model. It includes the inputs, commands, and questions to the model.
ipython	A new role introduced in Llama 3.1. Semantically, this role means "tool". This role is used to mark messages with the output of a tool call when sent back to the model from the executor.
assistant	Represents the response generated by the AI model based on the context provided in the **system**, **ipython** and **user** prompts.

LLM Reasoning

❖ Llama 3.2 Prompt Template

Special Tokens	Description
<|begin_of_text|>	Specifies the start of the prompt.
<|end_of_text|>	Model will cease to generate more tokens. This token is generated only by the base models.
<|finetune_right_pad_id|>	This token is used for padding text sequences to the same length in a batch.
<|start_header_id|>	These tokens enclose the role for a particular message. The possible roles are: [system, user, assistant, and ipython]
<|end_header_id|>	
<|eom_id|>	End of message. A message represents a possible stopping point for execution where the model can inform the executor that a tool call needs to be made. This is used for multi-step interactions between the model and any available tools. This token is emitted by the model when the Environment: ipython instruction is used in the system prompt, or if the model calls for a built-in tool.
<|eot_id|>	End of turn. Represents when the model has determined that it has finished interacting with the user message that initiated its response. This is used in two scenarios: • at the end of a direct interaction between the model and the user • at the end of multiple interactions between the model and any available tools This token signals to the executor that the model has finished generating a response.
<|python_tag|>	Special tag used in the model's response to signify a tool call.