Tool Use

CS6960 MultiModal LLM Agents

Kenneth Marino

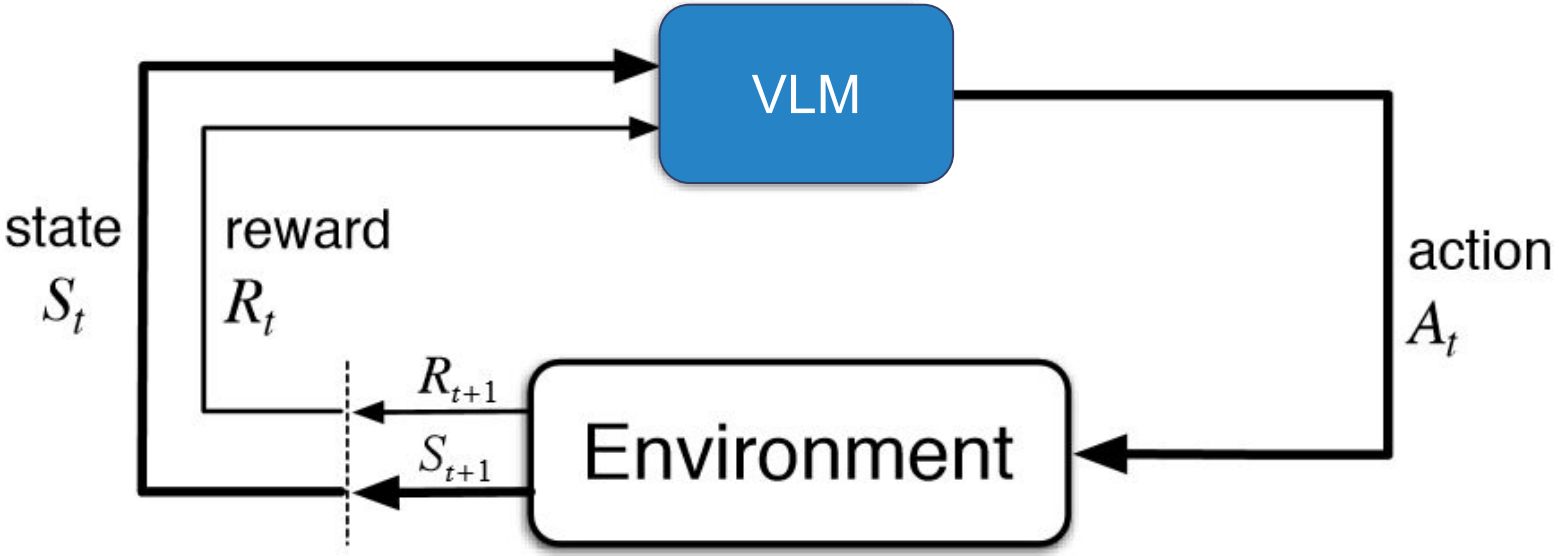
Slide credits: Graham Neubig, Hanna Hajishirzi, others

Announcements

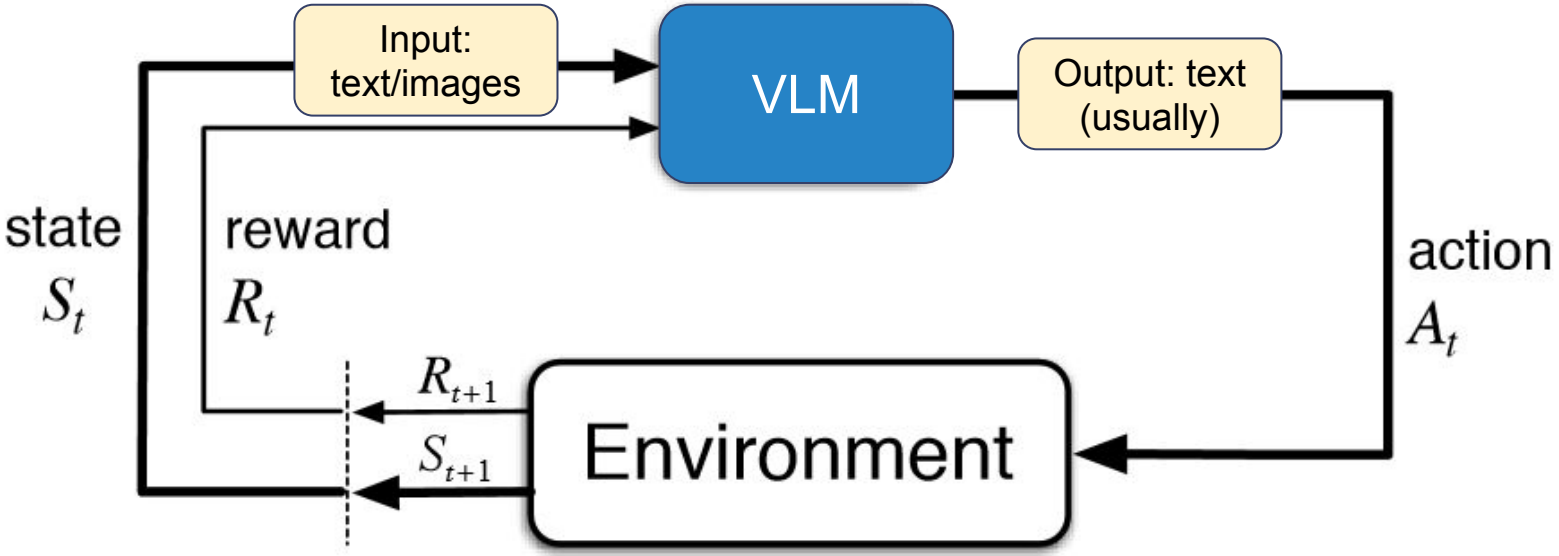
- HW2 due in 3 weeks
  - A bit more complicated than past homeworks
  - Have some extra time
- HW1
  - Still a few days left with late days+extension
- Projects
  - (Soft) deadline for group formation - Feb 10
  - If you don’t have a group by then, please post on piazza w/ your project idea/interests to find groups (we can play a little matchmaking)
  - Participate in group activity Thur
  - Reach out to us if you have any questions/concerns

Any Questions

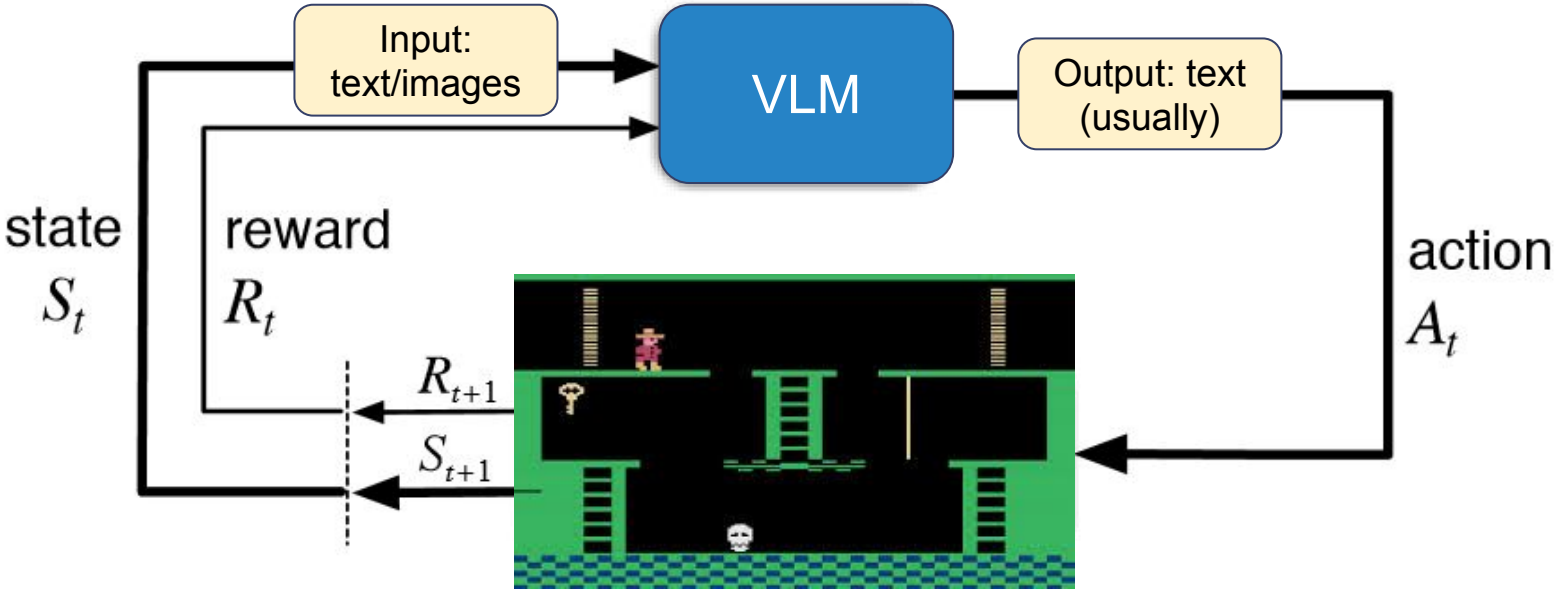
What does VLM agent actually look like?



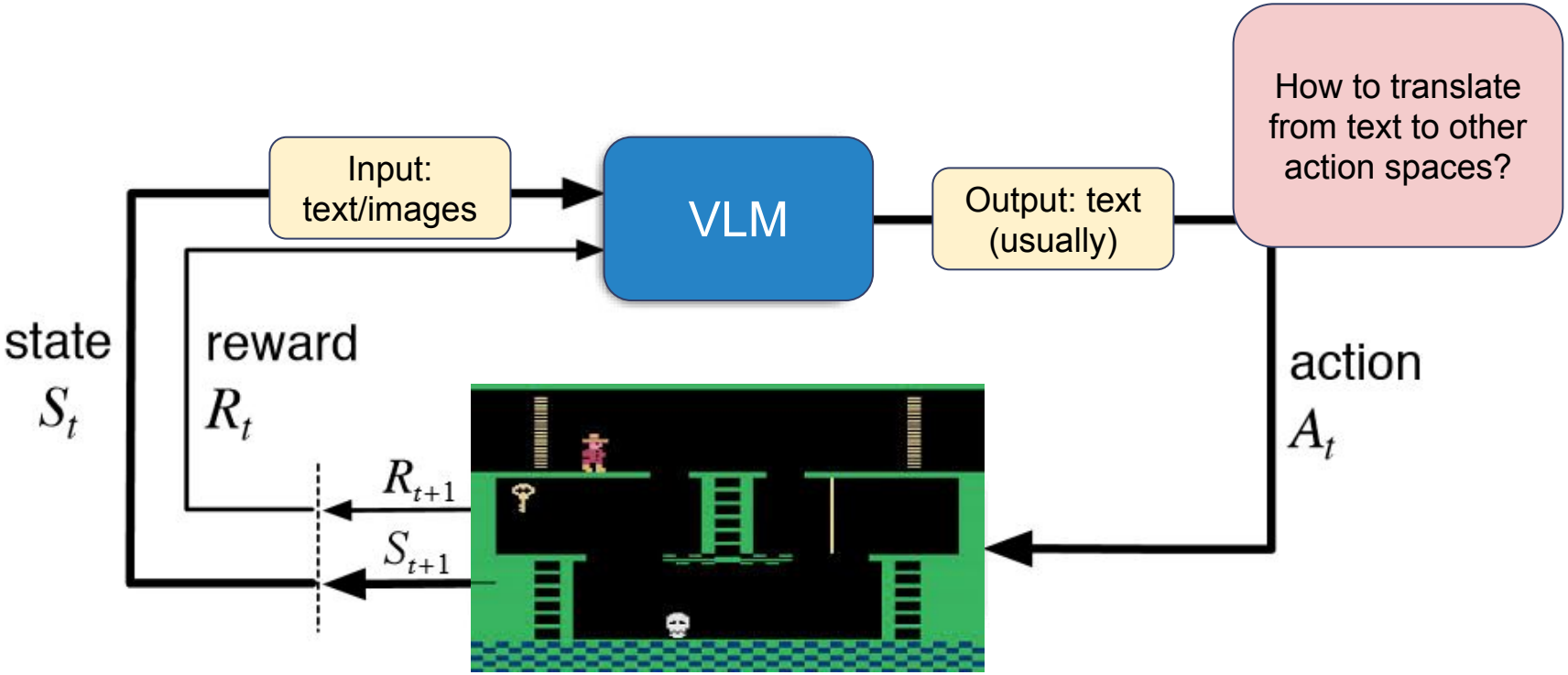
What does VLM agent actually look like?



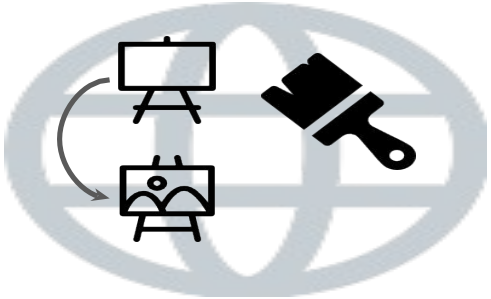
What does VLM agent actually look like?



What does VLM agent actually look like?



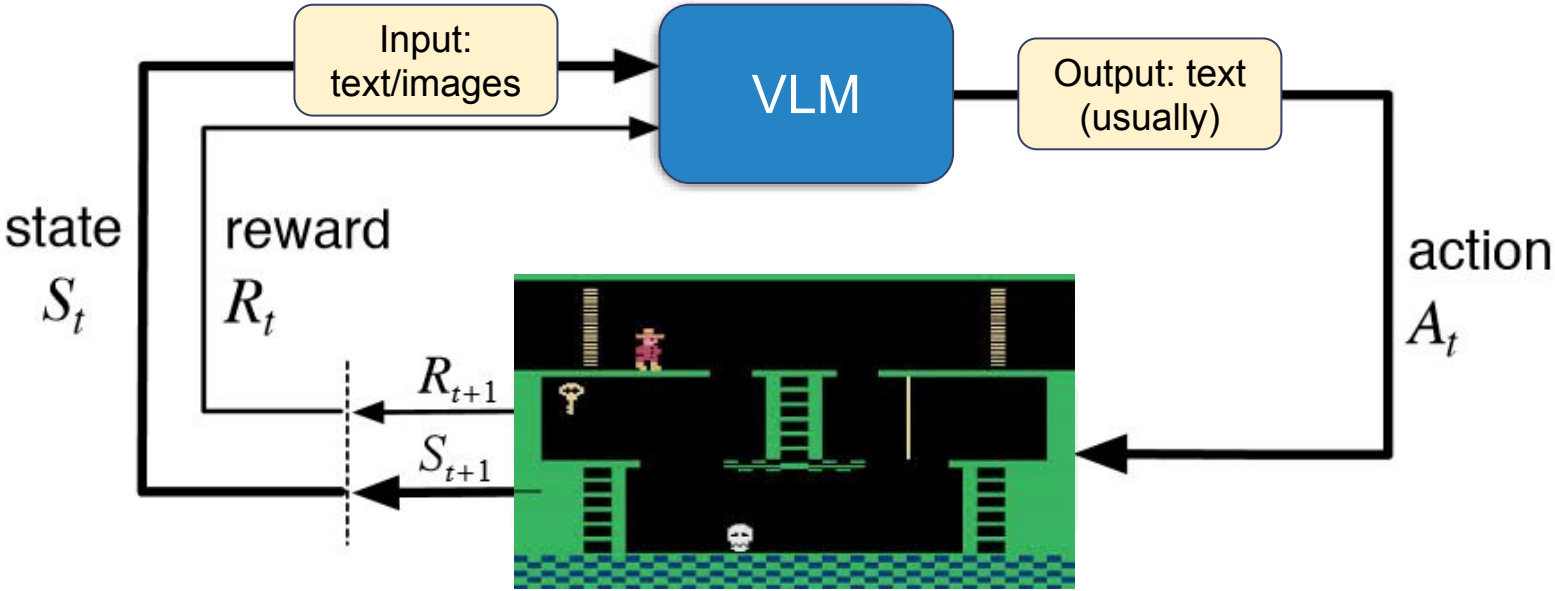
What is a Tool (for LLM Agents)?



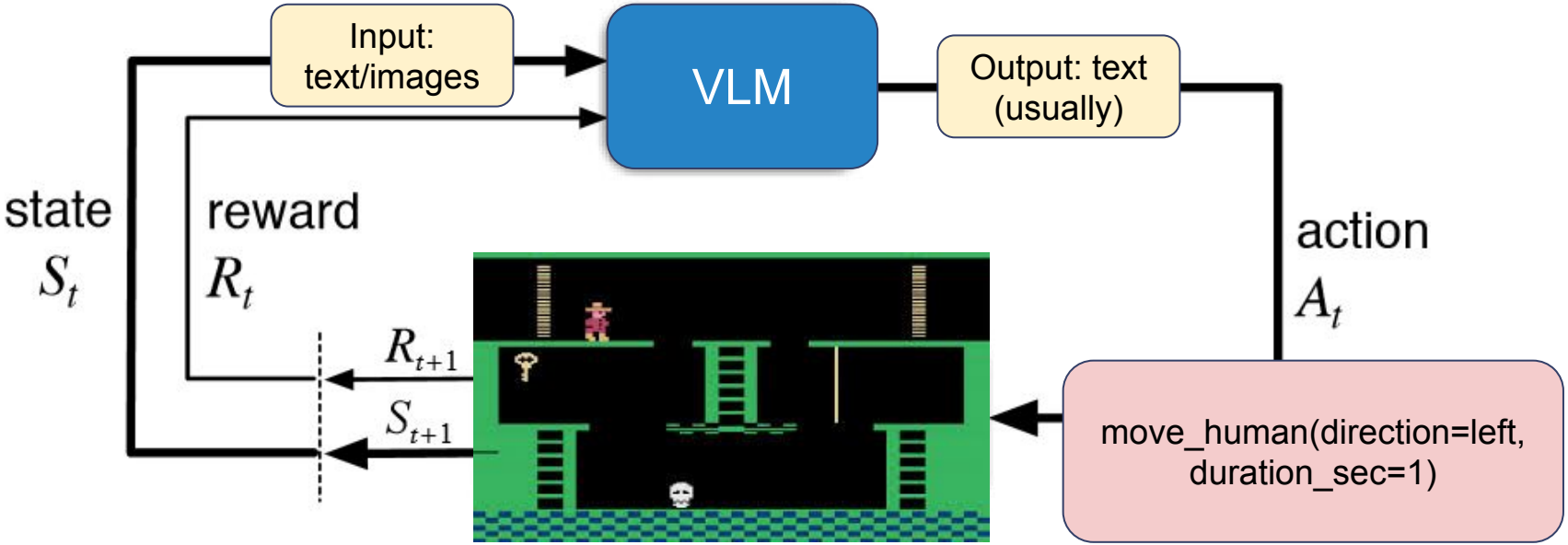
An LLM-used tool is a **function** interface to a computer program that runs **external** to the LLM, where the LLM **generates the function calls** and input **arguments** in order to use the tool.

Slide credit: Graham Neubig

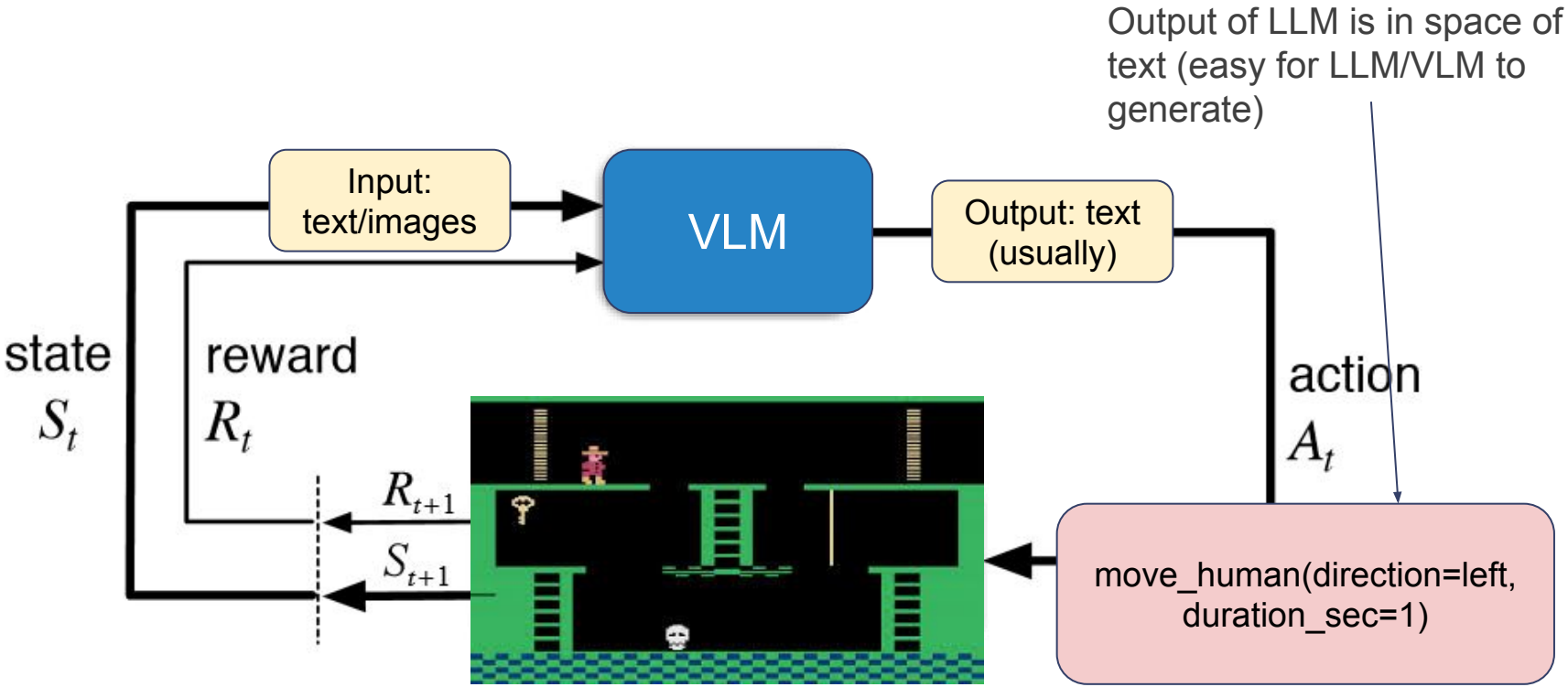
What is a Tool (for LLM Agents)?



What is a Tool (for LLM Agents)?

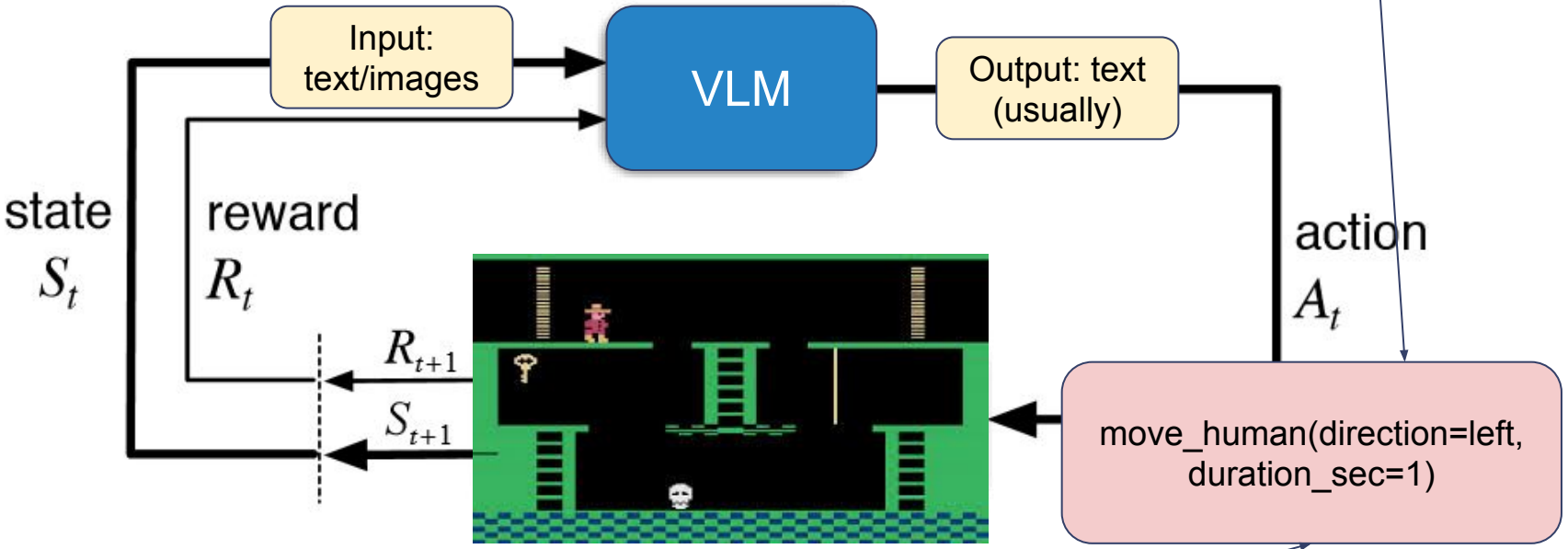


What is a Tool (for LLM Agents)?



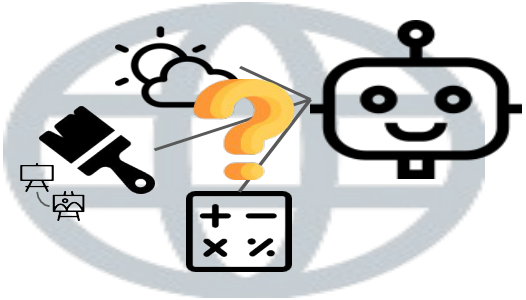
What is a Tool (for LLM Agents)?

Output of LLM is in space of text (easy for LLM/VLM to generate)



Tool call and parameters gives information needed by external tool to execute in the environment

Many Tool Functionalities



- 👀 Perception: collect data from the env
- Action: exert actions, change env state
- 🔢 Computation: general acts of computing

→ ***Tools***

Slide credit: Graham Neubig

Tool Use Scenarios

Category	Example Tools

Knowledge access	`sql_executor(query: str) -> answer: any`
	`search_engine(query: str) -> document: str`
	`retriever(query: str) -> document: str`

Computation activities	`calculator(formula: str) -> value: int | float`
	`python_interpreter(program: str) -> result: any`
	`worksheet.insert_row(row: list, index: int) -> None`

Interaction w/ the world	`get_weather(city_name: str) -> weather: str`
	`get_location(ip: str) -> location: str`
	`calendar.fetch_events(date: str) -> events: list`
	`email.verify(address: str) -> result: bool`

Non-textual modalities	`cat_image.delete(image_id: str) -> None`
	`spotify.play_music(name: str) -> None`
	`visual_qa(query: str, image: Image) -> answer: str`

Special-skilled LMs	`QA(question: str) -> answer: str` `translation(text: str, language: str) -> text: str`

Table 1: Exemplar tools for each category.

Slide credit: Graham Neubig

Tool Use Scenarios

Category	Example Tools

Knowledge access	`sql_executor(query: str) -> answer: any`
	`search_engine(query: str) -> document: str`
	`retriever(query: str) -> document: str`

Computation activities	`calculator(formula: str) -> value: int | float`
	`python_interpreter(program: str) -> result: any`
	`worksheet.insert_row(row: list, index: int) -> None`

Interaction w/ the world	`get_weather(city_name: str) -> weather: str`
	`get_location(ip: str) -> location: str`
	`calendar.fetch_events(date: str) -> events: list`
	`email.verify(address: str) -> result: bool`

Non-textual modalities	`cat_image.delete(image_id: str) -> None`
	`spotify.play_music(name: str) -> None`
	`visual_qa(query: str, image: Image) -> answer: str`

Special-skilled LMs	`QA(question: str) -> answer: str` `translation(text: str, language: str) -> text: str`

Table 1: Exemplar tools for each category.

Retrieval can be thought of as a tool

Slide credit: Graham Neubig

Recall: Retrieval as an agent tool

Level 1

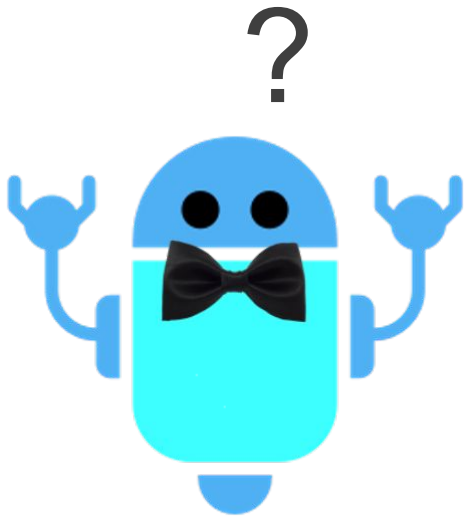
Question: What was the actual enrollment count of the clinical trial on H. pylori in acne vulgaris patients from Jan-May 2018 as listed on the NIH website?
Ground truth: 90

Level 2

Question: If this whole pint is made up of ice cream, how many percent above or below the US federal standards for butterfat content is it when using the standards as reported by Wikipedia in 2020? Answer as + or - a number rounded to one decimal place.
Ground truth: +4.6

Level 3

Question: In NASA's Astronomy Picture of the Day on 2006 January 21, two astronauts are visible, with one appearing much smaller than the other. As of August 2023, out of the astronauts in the NASA Astronaut Group that the smaller astronaut was a member of, which one spent the least time in space, and how many minutes did he spend in space, rounded to the nearest minute? Exclude any astronauts who did not spend any time in space. Give the last name of the astronaut, separated from the number of minutes by a semicolon. Use commas as thousands separators in the number of minutes.
Ground truth: White; 5876



Define Custom Tools for Env

move_agent()
jump()
check_inventory()
get_enemy_location()
get_item_location()

Other Models Can Be Tools

- call_vlm_detector()
- Returns:

person (50, 50, 100, 100)

skull (200, 100, 250, 150)

key (25, 100, 75, 150)

Other Models Can Be Tools

- planner_llm()
- Returns:

First you should jump across the gaps to….

Tool Use Paradigms

- Tool Use: switching between:
  - text-generation mode
  - tool-execution mode
- How to induce tool use
  - Inference-time prompting
  - Training

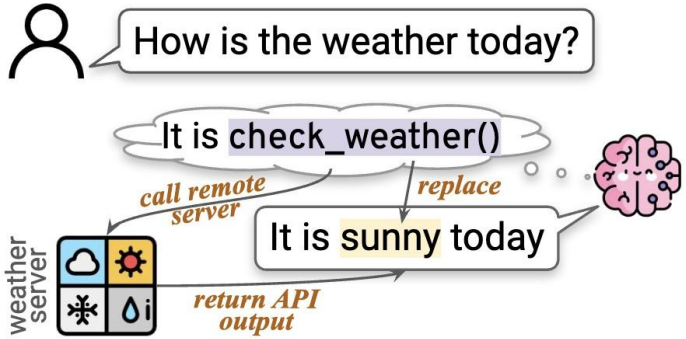
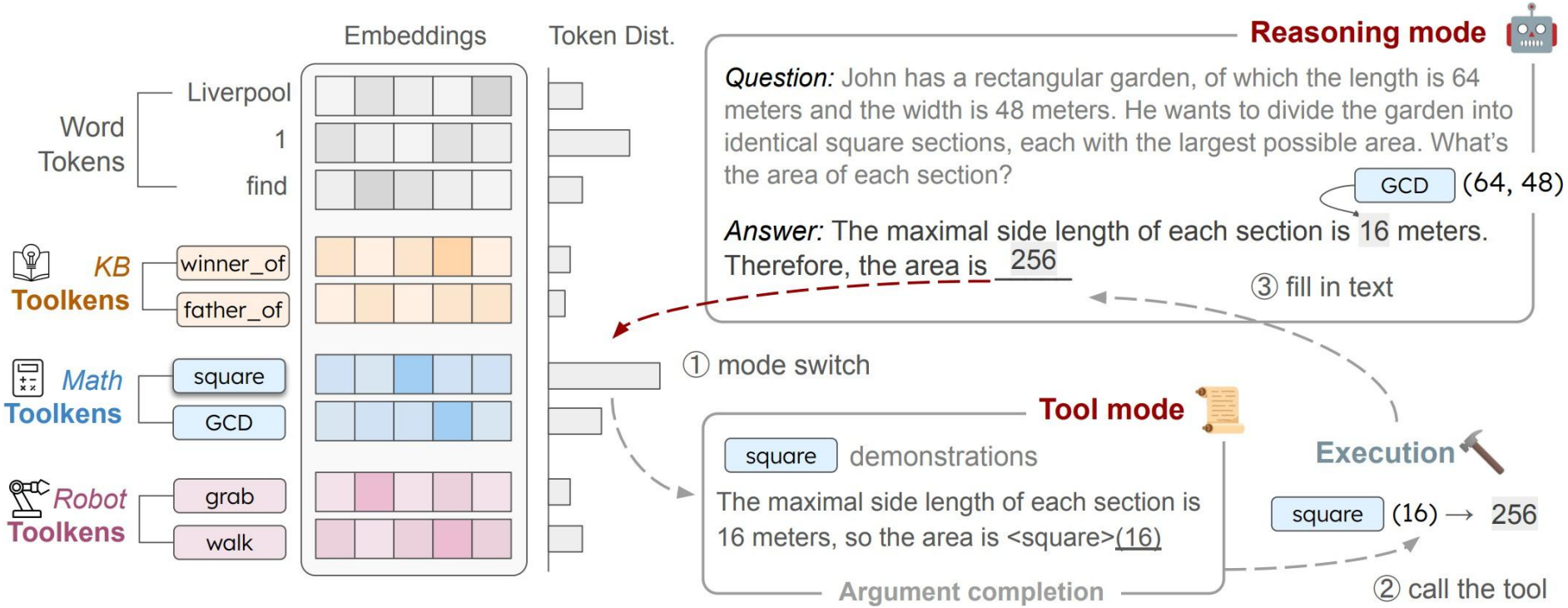


Figure 2: The basic tool use paradigm. LM calls `check_weather` tool by generating text tokens. This call triggers the server to execute the call and return the output sunny, using which the LM replaces the API call tokens in the response to the user.

Slide credit: Graham Neubig

Tool Execution: Tool Tokens



e.g. ToolkenGPT (Hao et al. 2023)

Slide credit: Graham Neubig

Tool Execution: Code Tags

I should calculate the phone price in USD for each country, then find the most cost-effective country.

<execute_python>

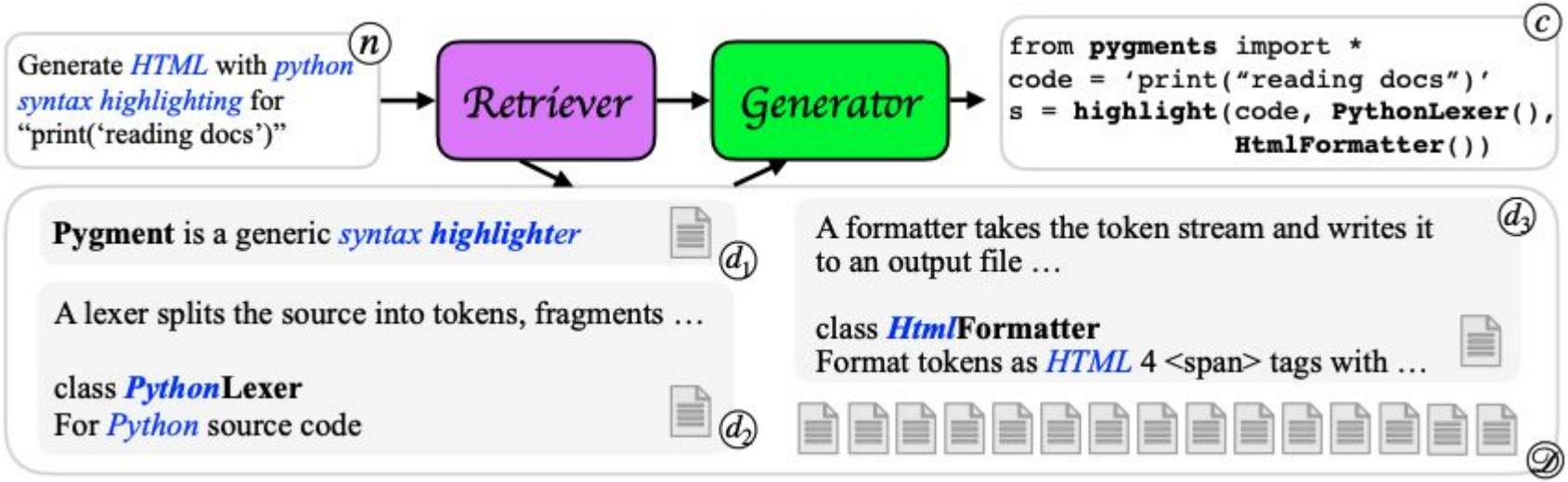
```
countries = ['USA', 'Japan', 'Germany', 'India']
final_prices = {}

for country in countries:
    exchange_rate, tax_rate = lookup_rates(country)
    local_price = lookup_phone_price("xAct 1", country)
    converted_price = convert_and_tax(
        local_price, exchange_rate, tax_rate
    )
    shipping_cost = estimate_shipping_cost(country)
    final_price = estimate_final_price(converted_price, shipping_cost)
    final_prices[country] = final_price

most_cost_effective_country = min(final_prices, key=final_prices.get)
most_cost_effective_price = final_prices[most_cost_effective_country]
print(most_cost_effective_country, most_cost_effective_price)
```

<execute_python>

e.g. ToolkenGPT (Hao et al. 2023)

Slide credit: Graham Neubig

Prompting for Tool Use



e.g. DocPrompting (Zhou et al. 2022) retrieves library documentation

Slide credit: Graham Neubig

Learning for Tool Use

Prompt Unsupervised

Your task is to add calls to a Question Answering API to a piece of text. The questions should help you get information required to complete the text. You can call the API by writing "[QA(question)]" where "question" is the question you want to ask. Here are some examples of API calls:

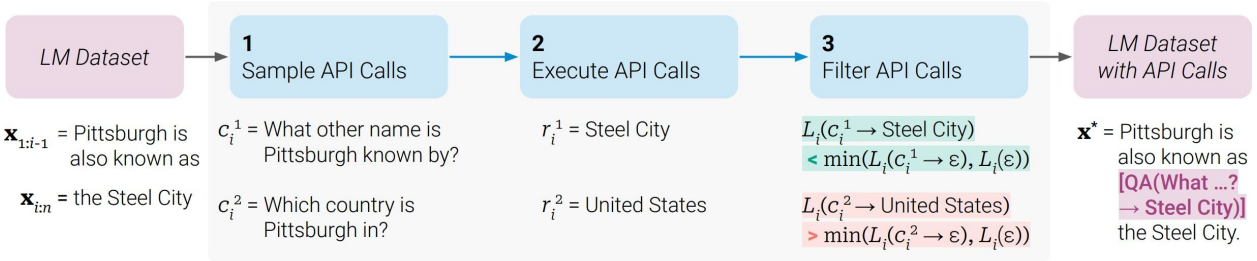
Input: Coca-Cola, or Coke, is a carbonated soft drink manufactured by the Coca-Cola Company.

Output: Coca-Cola, or [QA("What other name is Coca-Cola known by?")] Coke, is a carbonated soft drink manufactured by [QA("Who manufactures Coca-Cola?")] the Coca-Cola Company.

Input: $\mathbf{x}$

Output:

Filter for Success and Train



e.g. ToolFormer (Schick et al. 2023)

Slide credit: Graham Neubig

OpenAI Function Calling Standard

- Define a function signature in a Python dictionary

```
tools = [
  {
    "name": "get_delivery_date",
    "description": "Get the delivery date for a customer's order. Call this
whenever you need to know the delivery date, for example when a customer asks
'Where is my package'",
    "parameters": {
        "type": "object",
        "properties": {
            "order_id": {
                "type": "string",
                "description": "The customer's order ID."
            }
        },
        "required": ["order_id"],
        "additionalProperties": false
    }
  }
]
```

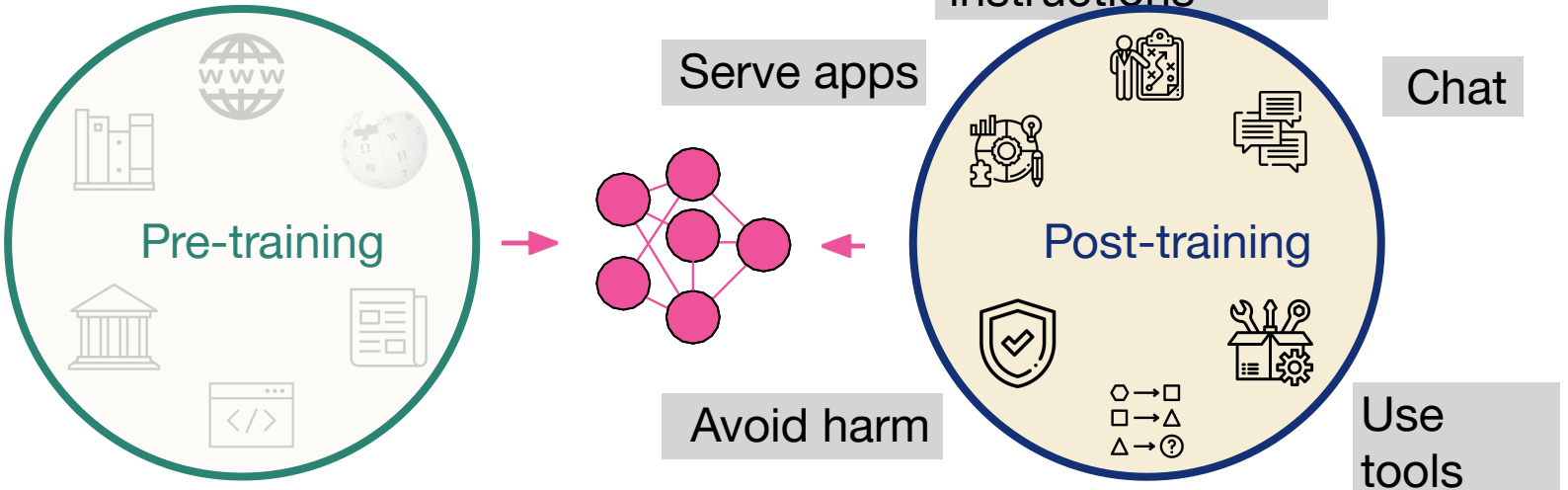
- Send it together with your prompt

```
response = openai.chat.completions.create(
    model="gpt-4o",
    messages=messages,
    tools=tools,
)
```

(OpenAI 2024)

Slide credit: Graham Neubig

Modern LLMs: trained to use tools



Predict the next word in various contexts

Slide credit: Hanna Hajishirzi

Takeaways

- Retrieval important part of agents
- Tools
  - A way of going from language to *some other* space
  - Can be used for taking actions, getting new input/observations, calling other models
  - A useful interface to the world
- Lots of ways of doing it
  - Using special tokens
  - Using code
- How to get to work
  - Pure prompting
  - Trained
  - Virtually all LLMs/VLMs trained to use tools in some way

Any Questions

Questions

Now for the presentations!