

AutoGPT

Basic usage

Aleš Kalfas

June 10, 2024 v1

Setup

1. Fork repo

- <https://github.com/Significant-Gravitas/AutoGPT>
 - 43k forks ★

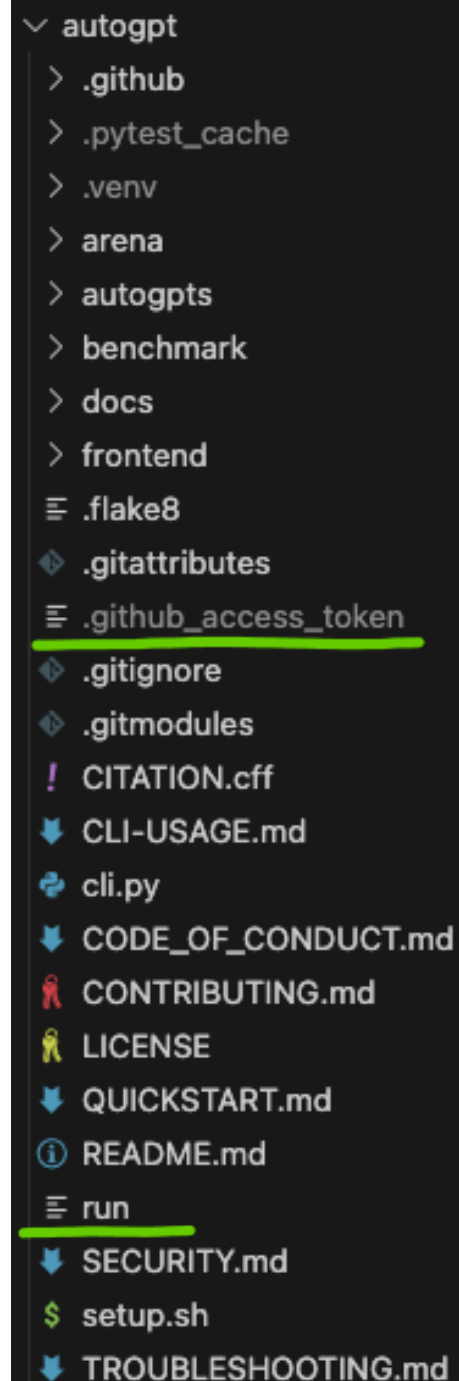
2. Clone forked repo

3. Insert github access token to the file

- *.github_access_token* file
- Token is using for leaderboard PR creation

4. Run CLI tool

```
./run setup // CLI tool helps setup dependencies and  
            manage agents  
  
            // Install basic dependencies (python3, poetry)
```



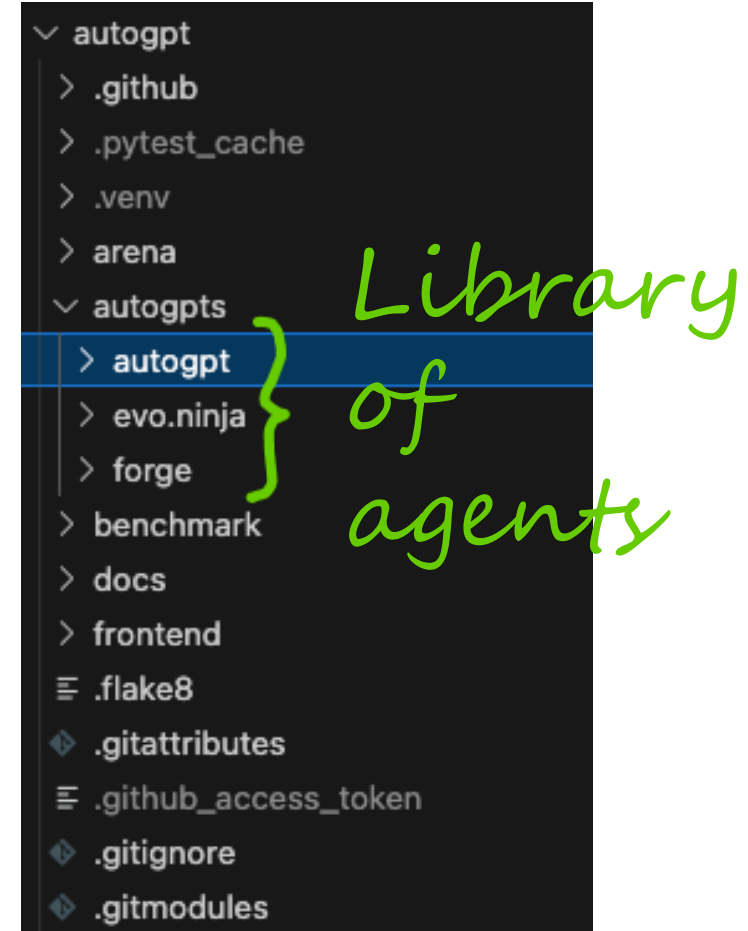
```
▼ autogpt  
  > .github  
  > .pytest_cache  
  > .venv  
  > arena  
  > autogpts  
  > benchmark  
  > docs  
  > frontend  
  ≡ .flake8  
  ◆ .gitattributes  
  ≡ .github_access_token  
  ◆ .gitignore  
  ◆ .gitmodules  
  ! CITATION.cff  
  ↓ CLI-USAGE.md  
  🔄 cli.py  
  ↓ CODE_OF_CONDUCT.md  
  🚫 CONTRIBUTING.md  
  📜 LICENSE  
  ↓ QUICKSTART.md  
  ⓘ README.md  
  ≡ run  
  ↓ SECURITY.md  
  $ setup.sh  
  ↓ TROUBLESHOOTING.md
```

Agent run

- Use an agent from the library
 - *autogpt*
 - the reference implementation
 - *evo.ninja*
 - <https://github.com/polywrap/evo.ninja>
 - *forge*
 - boilerplate code for new agent

```
./run agent start autogpt
```

```
// Run autogpt agent
```



```
(.venv) → autogpt git:(my-playground) x ./run agent start autogpt
Running setup for agent 'autogpt'...
Installing dependencies from lock file

No dependencies to install or update

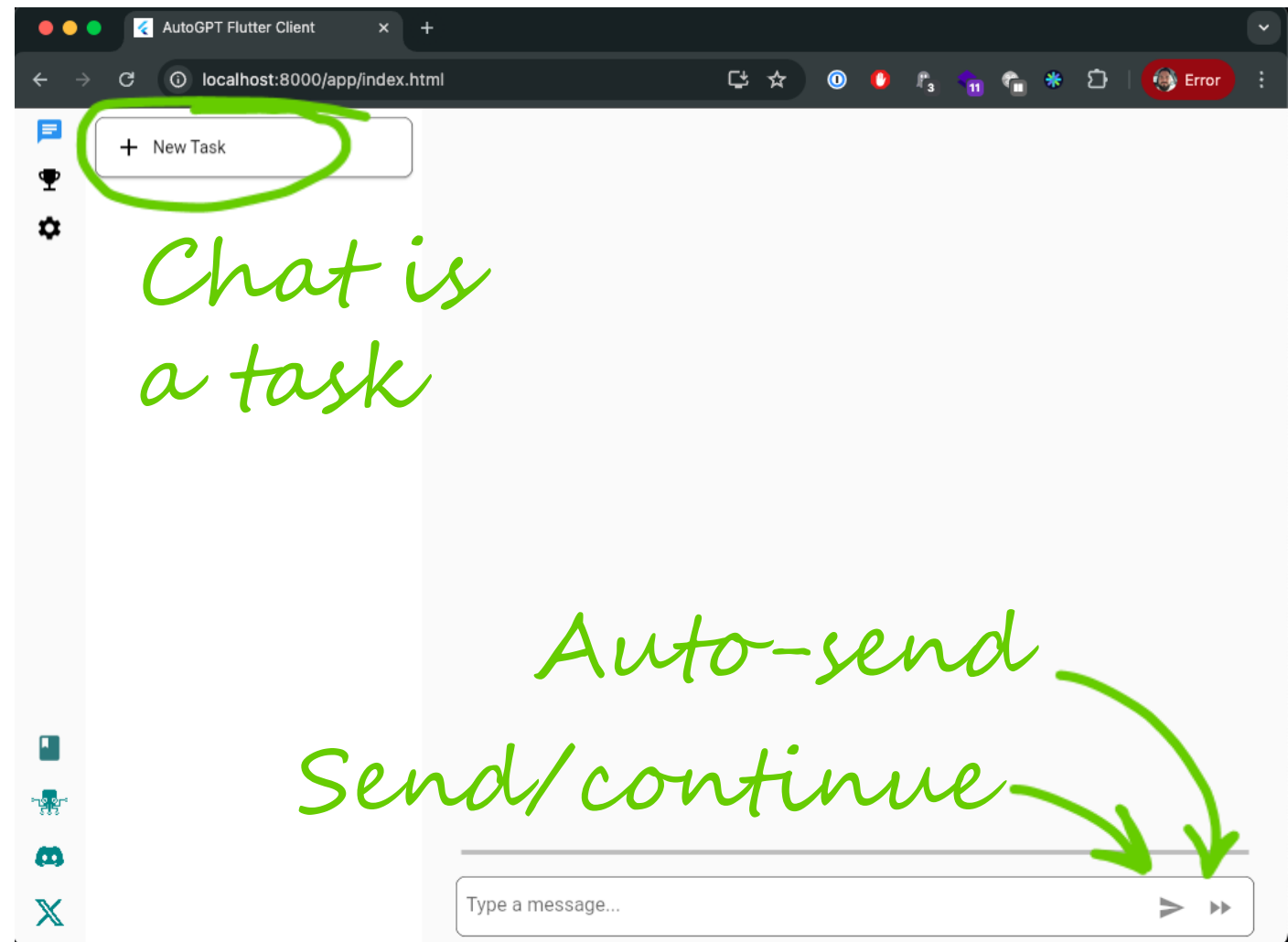
Installing the current project: agpt (0.5.0)
Setup completed successfully.

(Restarting benchmark server...
INFO: Started server process [72344]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8080 (Press CTRL+C to quit)

(Restarting agent 'autogpt'...
kill: usage: kill [-s sigspec | -n signum | -sigspec] pid | jobspec ... or kill -l [sigspec]
2024-06-11 07:38:44,676 INFO HTTP Request GET https://api.openai.com/v1/models "HTTP/1.1 200 OK"
2024-06-11 07:38:45,019 INFO HTTP Request GET https://api.openai.com/v1/models "HTTP/1.1 200 OK"
2024-06-11 07:38:45,061 INFO AutoGPT server starting on http://localhost:8000
Agent application started and available on port 8000
```

Setup
Benchmark server

Agent server



Chat is
a task

Auto-send
Send/continue



+ New Task

Hi





+ New Task

Hi!

User Hi!

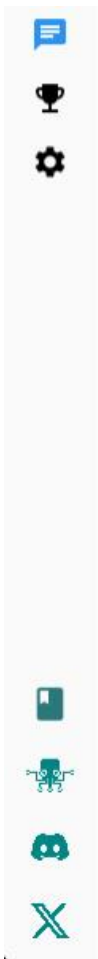
Agent Washington D.C

0 Artifacts



Type a message...





+ New Task

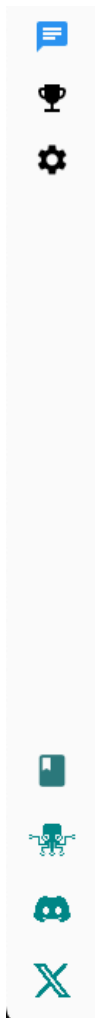
Hi!

User Hi!

Agent Washington D.C 0 Artifacts ^

```
{
  "name": "Hi!",
  "input": "Hi!",
  "additional_input": {},
  "created_at": "2024-06-10T17:23:20.951792",
  "modified_at": "2024-06-10T17:23:20.951796",
  "task_id": "e045294d-577c-45e1-8493-ce71b18951a3",
  "step_id": "bb4532fe-c11d-440d-b670-8e3b31831643",
  "status": "created",
  "output": "Washington D.C",
  "additional_output": null
}
```

Type a message... > >>



+ New Task

Who is the president of the US?

User Who is the president of the US?

Output

Next
command

Continue

Type a message... > >>

===== ChatPrompt =====

Length: 5 messages

----- SYSTEM -----

You are InfoGPT, an information retrieval AI designed to provide accurate and up-to-date facts about current world leaders and political figures.

Your decisions must always be made independently without seeking user assistance. Play to your strengths as an LLM and pursue simple strategies with no legal complications.

Constraints

You operate within the following constraints:

1. Exclusively use the commands listed below.
2. You can only act proactively, and are unable to start background jobs or set up webhooks for yourself. Take this into account when planning your actions.
3. You are unable to interact with physical objects. If this is absolutely necessary to fulfill a task or objective or to complete a step, you must ask the user to do so for you. If this is not possible, you must abort the task.
4. Do not provide unverified or speculative information.
5. Limit responses to factual information, avoiding opinions or biases.
6. Ensure privacy and data protection standards are upheld when handling user inquiries.

Resources

You can leverage access to the following resources:

1. Internet access for searches and information gathering.
2. The ability to read and write files.
3. You are a Large Language Model, trained on millions of pages of text, including a lot of factual knowledge. Make use of this factual knowledge to avoid unnecessary gathering of information.

Commands

These are the ONLY commands you can use. Any action you perform must be possible through one of these commands:

1. execute_python_code: Executes the given Python code inside a single-use Docker container with access to your workspace folder. Params: (code: string)
2. execute_python_file: Execute an existing Python file inside a single-use Docker container with access to your workspace folder. Params: (filename: string, args?: Array<string>)
3. list_folder: List the items in a folder. Params: (folder: string)
4. open_file: Opens a file for editing or continued viewing; creates it if it does not exist yet. Note: If you only need to read or write a file once, use 'write to file' instead.. Params: (filename: string)
5. open_folder: Open a folder to keep track of its content. Params: (path: string)
6. read_file: Read an existing file. Params: (filename: string)
7. write_file: Write a file, creating it if necessary. If the file exists, it is overwritten.. Params: (filename: string, contents: string)
8. ask_user: If you need more details or information regarding the given goals, you can ask the user for input. Params: (question: string)
9. read_webpage: Read a webpage, and extract specific information from it. You must specify either topics_of_interest, a question, or get_raw_content.. Params: (url: string, topics_of_interest: string, question: string, get_raw_content: boolean)
10. finish: Use this to shut down once you have completed your task, or when there are insurmountable problems that make it impossible for you to finish your task.. Params: (reason: string)

Best practices

1. Continuously review and analyze your actions to ensure you are performing to the best of your abilities.
2. Constructively self-criticize your big-picture behavior constantly.
3. Reflect on past decisions and strategies to refine your approach.
4. Every command has a cost, so be smart and efficient. Aim to complete tasks in the least number of steps.
5. Only make use of your information gathering abilities to find information that you don't yet have knowledge of.
6. Provide trustworthy and verified information from reliable sources.
7. Present information in a clear and accessible format, making it easy for users to understand.
8. Stay up-to-date with the latest information on changes in political leadership.
9. Ensure factual accuracy through regular updates and corrections as needed.

Your Task

The user will specify a task for you to execute, in triple quotes, in the next message. Your job is to complete the task while following your directives as given above, and terminate the task once you are done.

----- USER -----

"""Who is the president of the US?"""

----- SYSTEM -----

The current time and date is Mon Jun 10 14:39:37 2024

----- SYSTEM -----

Respond with pure JSON. The JSON object should be compatible with the TypeScript type 'Response' from the following:

```
interface Response {  
  thoughts: {  
    // Relevant observations from your last action (if any)  
  };  
  observations: string;  
  // Thoughts  
  text: string;  
  reasoning: string;  
  // Constructive self-criticism  
  self_criticism: string;  
}
```

} Personality

} Constraints

} Abilities

} Commands

} Reasoning

The user input

} Response format

Tokens counts

```
2024-06-10 14:39:47,300 DEBUG _trace.py:85 response_closed.complete
2024-06-10 14:39:47,301 DEBUG openai.py:636 Completion usage: 1061 input, 226 output - $0.01739
2024-06-10 14:39:47,302 DEBUG one_shot.py:381 LLM response content:
```

```
{
  "thoughts": {
    "observations": "The user asked for the current President of the US.",
    "text": "To provide the most accurate and up-to-date information, it's best to verify the current president's name, as changes in leadership could have occurred since",
    "reasoning": "Checking the latest information ensures accuracy and relevancy, adhering to the guidelines of providing verified information.",
    "self_criticism": "Initially, I considered answering directly from my knowledge base, but realizing political positions could change rapidly, it's more prudent to verify",
    "plan": "* Fetch information about the current US president using a web resource \n* Verify the accuracy of the information",
    "speak": "Let me verify the current President of the United States to ensure the most accurate and up-to-date information."
  },
  "command": {
    "name": "read_webpage",
    "args": {
      "url": "https://www.whitehouse.gov/",
      "topics_of_interest": ["current president"]
    }
  }
}
```

LLM response

```
2024-06-10 14:39:47,304 DEBUG one_shot.py:390 Validating object extracted from LLM response:
```

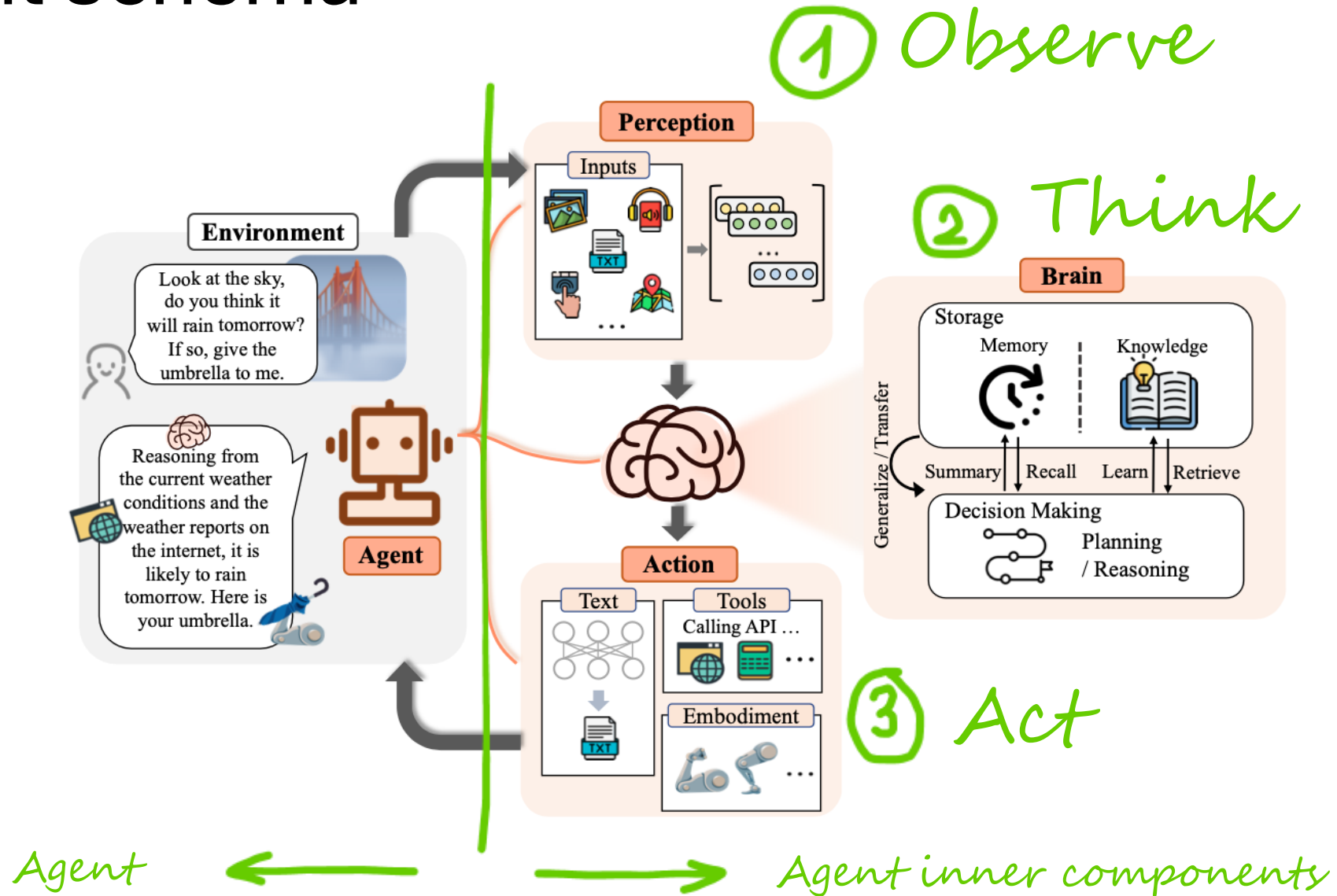
```
{
  "thoughts": {
    "observations": "The user asked for the current President of the US.",
    "text": "To provide the most accurate and up-to-date information, it's best to verify the current president's name, as changes in leadership could have occurred",
    "reasoning": "Checking the latest information ensures accuracy and relevancy, adhering to the guidelines of providing verified information.",
    "self_criticism": "Initially, I considered answering directly from my knowledge base, but realizing political positions could change rapidly, it's more prudent",
    "plan": "* Fetch information about the current US president using a web resource \n* Verify the accuracy of the information",
    "speak": "Let me verify the current President of the United States to ensure the most accurate and up-to-date information."
  },
  "command": {
    "name": "read_webpage",
    "args": {
      "url": "https://www.whitehouse.gov/",
      "topics_of_interest": [
        "current president"
      ]
    }
  }
}
```

Format validated response

```
2024-06-10 14:39:47,305 DEBUG agent_protocol_server.py:291 AI output: {'thoughts': {'observations': 'The user asked for the current President of the US.', 'text': "To provide the most accurate and up-to-date information, it's best to verify the current president's name, as changes in leadership could have occurred since my last update.", 'reasoning': 'Checking the latest information ensures accuracy and relevancy, adhering to the guidelines of providing verified information.', 'self_criticism': 'Initially, I considered answering directly from my knowledge base, but realizing political positions could change rapidly, it's more prudent to verify from a current source.', 'plan': '* Fetch information about the current US president using a web resource \n* Verify the accuracy of the information', 'speak': 'Let me verify the current President of the United States to ensure the most accurate and up-to-date information.'}, 'command': {'name': 'read_webpage', 'args': {'url': 'https://www.whitehouse.gov/', 'topics_of_interest': ['current president']}}}
2024-06-10 14:39:47,306 DEBUG agent_protocol_server.py:347 Running total LLM cost for task fca456d5-f8b9-44d2-ac96-23b872045c1b: $0.026
[2024-06-10 14:39:47,306] [forge.sdk.db ] [DEBUG] Updating step with task_id: fca456d5-f8b9-44d2-ac96-23b872045c1b and step_id: 887c995c-dbe3-41d9-977d-415ac3020
```

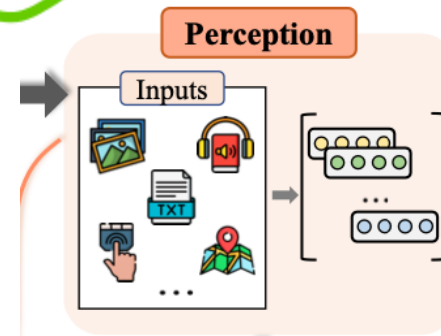
Tokens costs

Agent schema



One-shot prompt agent

① Observe

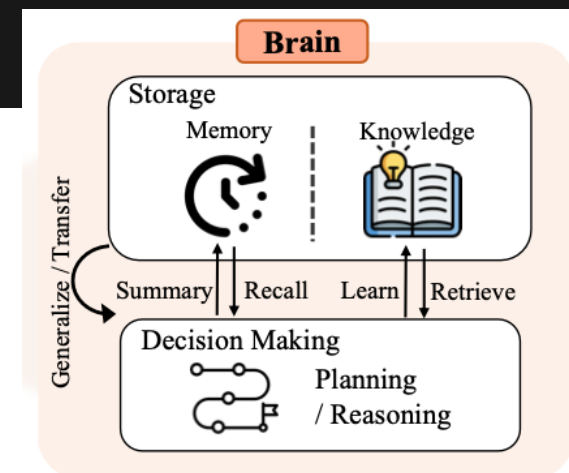


One shot prompt strategy

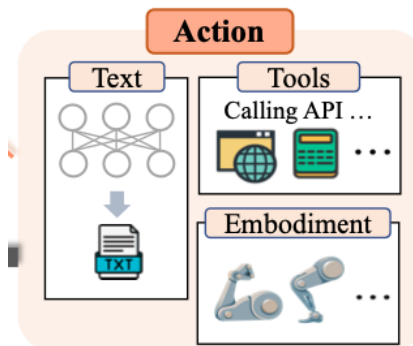
2024-06-10 14:39:47,302 DEBUG one_shot.py:381 LLM response content:

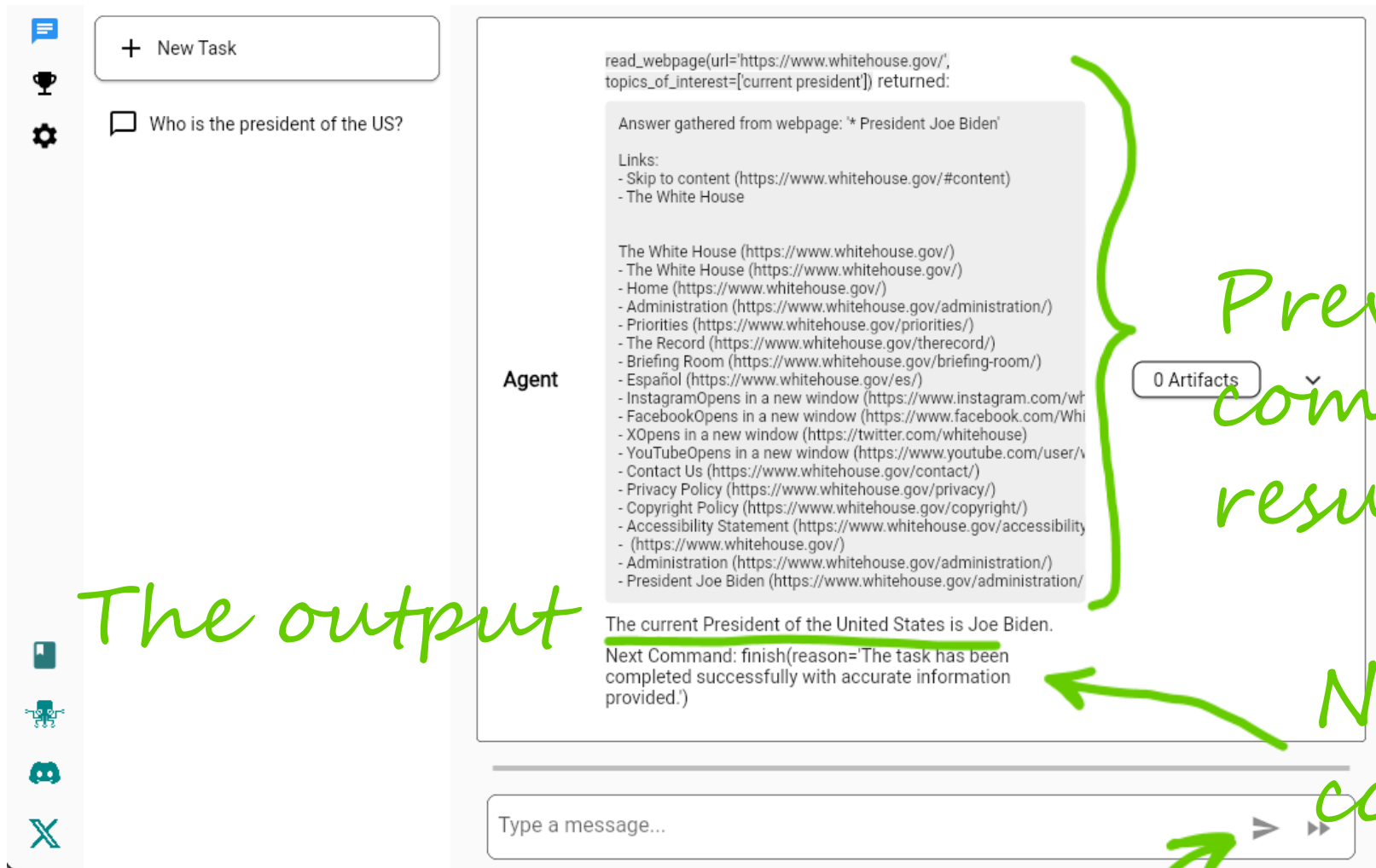
```
{
  "thoughts": {
    "observations": "The user asked for the current President of the US.",
    "text": "To provide the most accurate and up-to-date information, it's best to verify the current president's name, as changes in leadership could have occurred since my last update.",
    "reasoning": "Checking the latest information ensures accuracy and relevancy, adhering to the guidelines of providing verified information.",
    "self_criticism": "Initially, I considered answering directly from my knowledge base, but realizing political positions could change rapidly, it's more prudent to verify from a current source.",
    "plan": "* Fetch information about the current US president using a web resource \n* Verify the accuracy of the information",
    "peak": "Let me verify the current President of the United States to ensure the most accurate and up-to-date information."
  },
  "command": {
    "name": "read_webpage",
    "args": {
      "url": "https://www.whitehouse.gov/",
      "topics_of_interest": ["current president"]
    }
  }
}
```

② Think



③ Act





The output

Previous command result

Next command

Continue

Agent creation

```
./run agent create MyBestAgent
```

```
// Create an agent
```

The new agent



```
▼ autogpt
  > .github
  > .pytest_cache
  > .venv
  > arena
  ▼ autogpts
    > autogpt
    > evo.ninja
    > forge
  > benchmark
  > docs
  > frontend
  ≡ .flake8
  🔍 .gitattributes
  ≡ .github_access_token
  🔍 .gitignore
```

Library
of
agents

Agent folder structure

```
└─ autogpt
  └─ .github
  └─ .pytest_cache
  └─ .venv
  └─ arena
  └─ autogpts
    └─ autogpt
    └─ evo.ninja
    └─ forge
    └─ MyBestAgent
      └─ agbenchmark_config
      └─ forge
      └─ tutorials
      └─ __init__.py
      └─ .env.example
      └─ .flake8
      └─ .gitignore
      └─ .pre-commit-config.yaml
      └─ Dockerfile
      └─ mypy.ini
      └─ poetry.lock
      └─ pyproject.toml
      └─ README.md
      └─ run
      └─ run_benchmark
      └─ setup
      └─ benchmark
      └─ docs
```

The benchmark config
The main logic
Tutorials

Project technical setup

Helper scripts to control
from the parent CLI tool

The End

Problems

- Master branch doesn't work it is refactoring
- Last release 0.5.1
 - Custom agent setup file remove parent .venv environment!
 - error: externally-managed-environment
 - Doesn't work with python 3.12.X just 3.11.X
- *autogpt* reference implementation doesn't reflect current forge version
- Leaderboard doesn't work