OpenAI Multi Functions Agent

This notebook showcases using an agent that uses the OpenAI functions ability to respond to the prompts of the user using a Large Language Model

Install openai,google-search-results packages which are required as the langchain packages call them internally

> pip install openai google-search-results

```
from langchain import SerpAPIWrapper
from langchain.agents import initialize_agent, Tool
from langchain.agents import AgentType
from langchain.chat_models import ChatOpenAI
```

API Reference:

- initialize_agent from `langchain.agents`
- Tool from `langchain.agents`
- AgentType from `langchain.agents`
- ChatOpenAI from `langchain.chat_models`

The agent is given ability to perform search functionalities with the respective tool

SerpAPIWrapper:

> This initializes the SerpAPIWrapper for search functionality (search).

```
import getpass
import os

os.environ["SERPAPI_API_KEY"] = getpass.getpass()
```

```
    ........
```

```
# Initialize the OpenAI language model
# Replace <your_api_key> in openai_api_key="<your_api_key>" with your
```

```
actual OpenAI key.
llm = ChatOpenAI(temperature=0, model="gpt-3.5-turbo-0613")

# Initialize the SerpAPIWrapper for search functionality
# Replace <your_api_key> in openai_api_key="<your_api_key>" with your
actual SerpAPI key.
search = SerpAPIWrapper()

# Define a list of tools offered by the agent
tools = [
    Tool(
        name="Search",
        func=search.run,
        description="Useful when you need to answer questions about
current events. You should ask targeted questions.",
    ),
]
```

```
mrkl = initialize_agent(
    tools, llm, agent=AgentType.OPENAI_MULTI_FUNCTIONS, verbose=True
)
```

```
# Do this so we can see exactly what's going on under the hood
import langchain

langchain.debug = True
```

```
mrkl.run("What is the weather in LA and SF?")
```

```
    [chain/start] [1:chain:AgentExecutor] Entering Chain run with
input:
    {
      "input": "What is the weather in LA and SF?"
    }
    [llm/start] [1:chain:AgentExecutor > 2:llm:ChatOpenAI] Entering LLM
run with input:
    {
      "prompts": [
        "System: You are a helpful AI assistant.\nHuman: What is the
weather in LA and SF?"
      ]
    }
```

```
    [llm/end] [1:chain:AgentExecutor > 2:llm:ChatOpenAI] [2.91s]
Exiting LLM run with output:
    {
      "generations": [
        [
          {
            "text": "",
            "generation_info": null,
            "message": {
              "content": "",
              "additional_kwargs": {
                "function_call": {
                  "name": "tool_selection",
                  "arguments": "{\n  \"actions\": [\n    {\n
\"action_name\": \"Search\",\n      \"action\": {\n
\"tool_input\": \"weather in Los Angeles\"\n      }\n    },\n    {\n
\"action_name\": \"Search\",\n      \"action\": {\n
\"tool_input\": \"weather in San Francisco\"\n      }\n    }\n  ]\n}"
                }
              },
              "example": false
            }
          }
        ]
      ],
      "llm_output": {
        "token_usage": {
          "prompt_tokens": 81,
          "completion_tokens": 75,
          "total_tokens": 156
        },
        "model_name": "gpt-3.5-turbo-0613"
      },
      "run": null
    }
    [tool/start] [1:chain:AgentExecutor > 3:tool:Search] Entering Tool
run with input:
    "{'tool_input': 'weather in Los Angeles'}"
    [tool/end] [1:chain:AgentExecutor > 3:tool:Search] [608.693ms]
Exiting Tool run with output:
    "Mostly cloudy early, then sunshine for the afternoon. High 76F.
Winds SW at 5 to 10 mph. Humidity59%."
    [tool/start] [1:chain:AgentExecutor > 4:tool:Search] Entering Tool
run with input:
    "{'tool_input': 'weather in San Francisco'}"
    [tool/end] [1:chain:AgentExecutor > 4:tool:Search] [517.475ms]
Exiting Tool run with output:
```

```
    "Partly cloudy this evening, then becoming cloudy after midnight.
Low 53F. Winds WSW at 10 to 20 mph. Humidity83%."
    [llm/start] [1:chain:AgentExecutor > 5:llm:ChatOpenAI] Entering LLM
run with input:
    {
      "prompts": [
        "System: You are a helpful AI assistant.\nHuman: What is the
weather in LA and SF?\nAI: {'name': 'tool_selection', 'arguments':
'{\\n  \"actions\": [\\n    {\\n      \"action_name\": \"Search\",\\n
\"action\": {\\n        \"tool_input\": \"weather in Los Angeles\"\\n
}\\n    },\\n    {\\n      \"action_name\": \"Search\",\\n
\"action\": {\\n        \"tool_input\": \"weather in San Francisco\"\\n
}\\n    }\\n  ]\\n}'}\nFunction: Mostly cloudy early, then sunshine for
the afternoon. High 76F. Winds SW at 5 to 10 mph. Humidity59%.\nAI:
{'name': 'tool_selection', 'arguments': '{\\n  \"actions\": [\\n
{\\n      \"action_name\": \"Search\",\\n      \"action\": {\\n
\"tool_input\": \"weather in Los Angeles\"\\n      }\\n    },\\n
{\\n      \"action_name\": \"Search\",\\n      \"action\": {\\n
\"tool_input\": \"weather in San Francisco\"\\n      }\\n    }\\n
]\\n}'}\nFunction: Partly cloudy this evening, then becoming cloudy
after midnight. Low 53F. Winds WSW at 10 to 20 mph. Humidity83%."
      ]
    }
    [llm/end] [1:chain:AgentExecutor > 5:llm:ChatOpenAI] [2.33s]
Exiting LLM run with output:
    {
      "generations": [
        [
          {
            "text": "The weather in Los Angeles is mostly cloudy with a
high of 76°F and a humidity of 59%. The weather in San Francisco is
partly cloudy in the evening, becoming cloudy after midnight, with a
low of 53°F and a humidity of 83%.",
            "generation_info": null,
            "message": {
              "content": "The weather in Los Angeles is mostly cloudy
with a high of 76°F and a humidity of 59%. The weather in San Francisco
is partly cloudy in the evening, becoming cloudy after midnight, with a
low of 53°F and a humidity of 83%.",
              "additional_kwargs": {},
              "example": false
            }
          }
        ]
      ],
      "llm_output": {
        "token_usage": {
```

```
          "prompt_tokens": 307,
          "completion_tokens": 54,
          "total_tokens": 361
        },
        "model_name": "gpt-3.5-turbo-0613"
      },
      "run": null
    }
    [chain/end] [1:chain:AgentExecutor] [6.37s] Exiting Chain run with
output:
    {
      "output": "The weather in Los Angeles is mostly cloudy with a
high of 76°F and a humidity of 59%. The weather in San Francisco is
partly cloudy in the evening, becoming cloudy after midnight, with a
low of 53°F and a humidity of 83%."
    }
```

```
    'The weather in Los Angeles is mostly cloudy with a high of 76°F
and a humidity of 59%. The weather in San Francisco is partly cloudy in
the evening, becoming cloudy after midnight, with a low of 53°F and a
humidity of 83%.'
```

Configuring max iteration behavior

To make sure that our agent doesn't get stuck in excessively long loops, we can set max_iterations. We can also set an early stopping method, which will determine our agent's behavior once the number of max iterations is hit. By default, the early stopping uses method `force` which just returns that constant string. Alternatively, you could specify method `generate` which then does one FINAL pass through the LLM to generate an output.

```
mrkl = initialize_agent(
    tools,
    llm,
    agent=AgentType.OPENAI_FUNCTIONS,
    verbose=True,
    max_iterations=2,
    early_stopping_method="generate",
)
```

```
mrkl.run("What is the weather in NYC today, yesterday, and the day
before?")
```

```
    [chain/start] [1:chain:AgentExecutor] Entering Chain run with
input:
    {
      "input": "What is the weather in NYC today, yesterday, and the
day before?"
    }
    [llm/start] [1:chain:AgentExecutor > 2:llm:ChatOpenAI] Entering LLM
run with input:
    {
      "prompts": [
        "System: You are a helpful AI assistant.\nHuman: What is the
weather in NYC today, yesterday, and the day before?"
      ]
    }
    [llm/end] [1:chain:AgentExecutor > 2:llm:ChatOpenAI] [1.27s]
Exiting LLM run with output:
    {
      "generations": [
        [
          {
            "text": "",
            "generation_info": null,
            "message": {
              "lc": 1,
              "type": "constructor",
              "id": [
                "langchain",
                "schema",
                "messages",
                "AIMessage"
              ],
              "kwargs": {
                "content": "",
                "additional_kwargs": {
                  "function_call": {
                    "name": "Search",
                    "arguments": "{\n  \"query\": \"weather in NYC
today\"\n}"
                  }
                }
              }
            }
```

```
            }
          ]
        ],
        "llm_output": {
          "token_usage": {
            "prompt_tokens": 79,
            "completion_tokens": 17,
            "total_tokens": 96
          },
          "model_name": "gpt-3.5-turbo-0613"
        },
        "run": null
      }
      [tool/start] [1:chain:AgentExecutor > 3:tool:Search] Entering Tool
run with input:
      "{'query': 'weather in NYC today'}"
      [tool/end] [1:chain:AgentExecutor > 3:tool:Search] [3.84s] Exiting
Tool run with output:
      "10:00 am · Feels Like85° · WindSE 4 mph · Humidity78% · UV Index3
of 11 · Cloud Cover81% · Rain Amount0 in ..."
      [llm/start] [1:chain:AgentExecutor > 4:llm:ChatOpenAI] Entering LLM
run with input:
      {
        "prompts": [
          "System: You are a helpful AI assistant.\nHuman: What is the
weather in NYC today, yesterday, and the day before?\nAI: {'name':
'Search', 'arguments': '{\\n  \"query\": \"weather in NYC
today\"\\n}'}\nFunction: 10:00 am · Feels Like85° · WindSE 4 mph ·
Humidity78% · UV Index3 of 11 · Cloud Cover81% · Rain Amount0 in ..."
        ]
      }
      [llm/end] [1:chain:AgentExecutor > 4:llm:ChatOpenAI] [1.24s]
Exiting LLM run with output:
      {
        "generations": [
          [
            {
              "text": "",
              "generation_info": null,
              "message": {
                "lc": 1,
                "type": "constructor",
                "id": [
                  "langchain",
                  "schema",
                  "messages",
                  "AIMessage"
```

```
              ],
              "kwargs": {
                "content": "",
                "additional_kwargs": {
                  "function_call": {
                    "name": "Search",
                    "arguments": "{\n  \"query\": \"weather in NYC
yesterday\"\n}"
                  }
                }
              }
            }
          }
        ]
      ],
      "llm_output": {
        "token_usage": {
          "prompt_tokens": 142,
          "completion_tokens": 17,
          "total_tokens": 159
        },
        "model_name": "gpt-3.5-turbo-0613"
      },
      "run": null
    }
    [tool/start] [1:chain:AgentExecutor > 5:tool:Search] Entering Tool
run with input:
    "{'query': 'weather in NYC yesterday'}"
    [tool/end] [1:chain:AgentExecutor > 5:tool:Search] [1.15s] Exiting
Tool run with output:
    "New York Temperature Yesterday. Maximum temperature yesterday: 81
°F (at 1:51 pm) Minimum temperature yesterday: 72 °F (at 7:17 pm)
Average temperature ..."
    [llm/start] [1:llm:ChatOpenAI] Entering LLM run with input:
    {
      "prompts": [
        "System: You are a helpful AI assistant.\nHuman: What is the
weather in NYC today, yesterday, and the day before?\nAI: {'name':
'Search', 'arguments': '{\\n  \"query\": \"weather in NYC
today\"\\n}'}\nFunction: 10:00 am · Feels Like85° · WindSE 4 mph ·
Humidity78% · UV Index3 of 11 · Cloud Cover81% · Rain Amount0 in
...\nAI: {'name': 'Search', 'arguments': '{\\n  \"query\": \"weather in
NYC yesterday\"\\n}'}\nFunction: New York Temperature Yesterday.
Maximum temperature yesterday: 81 °F (at 1:51 pm) Minimum temperature
yesterday: 72 °F (at 7:17 pm) Average temperature ..."
      ]
    }
```

```
    [llm/end] [1:llm:ChatOpenAI] [2.68s] Exiting LLM run with output:
    {
      "generations": [
        [
          {
            "text": "Today in NYC, the weather is currently 85°F with a
southeast wind of 4 mph. The humidity is at 78% and there is 81% cloud
cover. There is no rain expected today.\n\nYesterday in NYC, the
maximum temperature was 81°F at 1:51 pm, and the minimum temperature
was 72°F at 7:17 pm.\n\nFor the day before yesterday, I do not have the
specific weather information.",
            "generation_info": null,
            "message": {
              "lc": 1,
              "type": "constructor",
              "id": [
                "langchain",
                "schema",
                "messages",
                "AIMessage"
              ],
              "kwargs": {
                "content": "Today in NYC, the weather is currently 85°F
with a southeast wind of 4 mph. The humidity is at 78% and there is 81%
cloud cover. There is no rain expected today.\n\nYesterday in NYC, the
maximum temperature was 81°F at 1:51 pm, and the minimum temperature
was 72°F at 7:17 pm.\n\nFor the day before yesterday, I do not have the
specific weather information.",
                "additional_kwargs": {}
              }
            }
          }
        ]
      ],
      "llm_output": {
        "token_usage": {
          "prompt_tokens": 160,
          "completion_tokens": 91,
          "total_tokens": 251
        },
        "model_name": "gpt-3.5-turbo-0613"
      },
      "run": null
    }
    [chain/end] [1:chain:AgentExecutor] [10.18s] Exiting Chain run with
output:
    {
```

```
      "output": "Today in NYC, the weather is currently 85°F with a
southeast wind of 4 mph. The humidity is at 78% and there is 81% cloud
cover. There is no rain expected today.\n\nYesterday in NYC, the
maximum temperature was 81°F at 1:51 pm, and the minimum temperature
was 72°F at 7:17 pm.\n\nFor the day before yesterday, I do not have the
specific weather information."
    }
```

```
    'Today in NYC, the weather is currently 85°F with a southeast wind
of 4 mph. The humidity is at 78% and there is 81% cloud cover. There is
no rain expected today.\n\nYesterday in NYC, the maximum temperature
was 81°F at 1:51 pm, and the minimum temperature was 72°F at 7:17
pm.\n\nFor the day before yesterday, I do not have the specific weather
information.'
```

Notice that we never get around to looking up the weather the day before yesterday, due to hitting our max_iterations limit.