LangChain: Agents

Outline:

- Using built in LangChain tools: DuckDuckGo search and Wikipedia
- Defining your own tools

In [18]:
```
import os

from dotenv import load_dotenv, find_dotenv
_ = load_dotenv(find_dotenv()) # read local .env file

import warnings
warnings.filterwarnings("ignore")
```

Built-in LangChain tools

In [19]:
```
#!pip install -U wikipedia
```

In [20]:
```
from langchain.agents.agent_toolkits import create_python_agent
from langchain.agents import load_tools, initialize_agent
from langchain.agents import AgentType
from langchain.tools.python.tool import PythonREPLTool
from langchain.python import PythonREPL
from langchain.chat_models import ChatOpenAI
```

In [21]:
```
llm = ChatOpenAI(temperature=0)
```

In [22]:
```
tools = load_tools(["llm-math","wikipedia"], llm=llm)
```

In [23]:
```
agent= initialize_agent(
    tools,
    llm,
    agent=AgentType.CHAT_ZERO_SHOT_REACT_DESCRIPTION,
    handle_parsing_errors=True,
    verbose = True)
```

In [24]:
```
agent("What is the 25% of 300?")
```

```
> Entering new AgentExecutor chain...
Thought: We need to calculate 25% of 300, which means we need to multiply 300
by 0.25.

Action:
```
{
  "action": "Calculator",
  "action_input": "300*0.25"
}
```


Observation: Answer: 75.0
Thought:The calculator tool returned the answer 75.0, which is 25% of 300.

Final Answer: 25% of 300 is 75.0.

> Finished chain.

{'input': 'What is the 25% of 300?', 'output': '25% of 300 is 75.0.'}
```

In [25]:
```
question = "Tom M. Mitchell is an American computer scientist \
and the Founders University Professor at Carnegie Mellon University (CMU)\
what book did he write?"
result = agent(question)
```

```
> Entering new AgentExecutor chain...
Thought: I should use Wikipedia to find the answer to this question.

Action:
```
{
  "action": "Wikipedia",
  "action_input": "Tom M. Mitchell"
}
```


Observation: Page: Tom M. Mitchell
Summary: Tom Michael Mitchell (born August 9, 1951) is an American computer s
cientist and the Founders University Professor at Carnegie Mellon University
(CMU). He is a founder and former Chair of the Machine Learning Department at
CMU. Mitchell is known for his contributions to the advancement of machine le
arning, artificial intelligence, and cognitive neuroscience and is the author
of the textbook Machine Learning. He is a member of the United States Nationa
l Academy of Engineering since 2010. He is also a Fellow of the American Acad
emy of Arts and Sciences, the American Association for the Advancement of Sci
ence and a Fellow and past President of the Association for the Advancement o
f Artificial Intelligence. In October 2018, Mitchell was appointed as the Int
erim Dean of the School of Computer Science at Carnegie Mellon.

Page: Tom Mitchell (Australian footballer)
Summary: Thomas Mitchell (born 31 May 1993) is a professional Australian rule
s footballer playing for the Collingwood Football Club in the Australian Foot
ball League (AFL). He previously played for the Sydney Swans from 2012 to 201
6, and the Hawthorn Football Club between 2017 and 2022. Mitchell won the Bro
wnlow Medal as the league's best and fairest player in 2018 and set the recor
d for the most disposals in a VFL/AFL match, accruing 54 in a game against Co
llingwood during that season.
Thought:The book that Tom M. Mitchell wrote is "Machine Learning".

Action:
```
{
  "action": "Wikipedia",
  "action_input": "Machine Learning (book)"
}
```


Observation: Page: Machine learning
Summary: Machine learning (ML) is a field devoted to understanding and buildi
ng methods that let machines "learn" – that is, methods that leverage data to
improve computer performance on some set of tasks.Machine learning algorithms
build a model based on sample data, known as training data, in order to make
predictions or decisions without being explicitly programmed to do so. Machin
e learning algorithms are used in a wide variety of applications, such as in
medicine, email filtering, speech recognition, agriculture, and computer visi
on, where it is difficult or unfeasible to develop conventional algorithms to
perform the needed tasks.A subset of machine learning is closely related to c
omputational statistics, which focuses on making predictions using computers,
```

```
but not all machine learning is statistical learning. The study of mathematic
al optimization delivers methods, theory and application domains to the field
of machine learning. Data mining is a related field of study, focusing on exp
loratory data analysis through unsupervised learning.Some implementations of
machine learning use data and artificial neural networks in a way that mimics
the working of a biological brain.In its application across business problem
s, machine learning is also referred to as predictive analytics.

Page: Quantum machine learning
Summary: Quantum machine learning is the integration of quantum algorithms wi
thin machine learning programs. The most common use of the term refers to mac
hine learning algorithms for the analysis of classical data executed on a qua
ntum computer, i.e. quantum-enhanced machine learning. While machine learning
algorithms are used to compute immense quantities of data, quantum machine le
arning utilizes qubits and quantum operations or specialized quantum systems
to improve computational speed and data storage done by algorithms in a progr
am. This includes hybrid methods that involve both classical and quantum proc
essing, where computationally difficult subroutines are outsourced to a quant
um device. These routines can be more complex in nature and executed faster o
n a quantum computer. Furthermore, quantum algorithms can be used to analyze
quantum states instead of classical data. Beyond quantum computing, the term
"quantum machine learning" is also associated with classical machine learning
methods applied to data generated from quantum experiments (i.e. machine lear
ning of quantum systems), such as learning the phase transitions of a quantum
system or creating new quantum experiments. Quantum machine learning also ext
ends to a branch of research that explores methodological and structural simi
larities between certain physical systems and learning systems, in particular
neural networks. For example, some mathematical and numerical techniques from
quantum physics are applicable to classical deep learning and vice versa. Fur
thermore, researchers investigate more abstract notions of learning theory wi
th respect to quantum information, sometimes referred to as "quantum learning
theory".

Page: Timeline of machine learning
Summary: This page is a timeline of machine learning. Major discoveries, achi
evements, milestones and other major events in machine learning are included.
Thought:Tom M. Mitchell wrote the book "Machine Learning".

Final Answer: "Machine Learning".

> Finished chain.
```

Python Agent

In [26]:
```python
agent = create_python_agent(
    llm,
    tool=PythonREPLTool(),
    verbose=True
)
```

In [27]:
```
customer_list = [["Harrison", "Chase"],
                 ["Lang", "Chain"],
                 ["Dolly", "Too"],
                 ["Elle", "Elem"],
                 ["Geoff","Fusion"],
                 ["Trance","Former"],
                 ["Jen","Ayai"]
                ]
```

In [28]:
```
agent.run(f"""Sort these customers by \
last name and then first name \
and print the output: {customer_list}""")
```

```
> Entering new AgentExecutor chain...
I can use the sorted() function to sort the list of customers by last name an
d then first name. I will need to provide a key function to sorted() that ret
urns a tuple of the last name and first name in that order.
Action: Python REPL
Action Input:
```
```
customers = [['Harrison', 'Chase'], ['Lang', 'Chain'], ['Dolly', 'Too'], ['El
le', 'Elem'], ['Geoff', 'Fusion'], ['Trance', 'Former'], ['Jen', 'Ayai']]
sorted_customers = sorted(customers, key=lambda x: (x[1], x[0]))
for customer in sorted_customers:
    print(customer)
```
```
Observation: ['Jen', 'Ayai']
['Lang', 'Chain']
['Harrison', 'Chase']
['Elle', 'Elem']
['Trance', 'Former']
['Geoff', 'Fusion']
['Dolly', 'Too']

Thought:The customers have been sorted by last name and then first name.
Final Answer: [['Jen', 'Ayai'], ['Lang', 'Chain'], ['Harrison', 'Chase'], ['E
lle', 'Elem'], ['Trance', 'Former'], ['Geoff', 'Fusion'], ['Dolly', 'Too']]

> Finished chain.

"[['Jen', 'Ayai'], ['Lang', 'Chain'], ['Harrison', 'Chase'], ['Elle', 'Ele
m'], ['Trance', 'Former'], ['Geoff', 'Fusion'], ['Dolly', 'Too']]"
```

View detailed outputs of the chains

In [29]:
```
import langchain
langchain.debug=True
agent.run(f"""Sort these customers by \
last name and then first name \
and print the output: {customer_list}""")
langchain.debug=False
```

```
[chain/start] [1:chain:AgentExecutor] Entering Chain run with input:
{
  "input": "Sort these customers by last name and then first name and prin
t the output: [['Harrison', 'Chase'], ['Lang', 'Chain'], ['Dolly', 'Too'],
['Elle', 'Elem'], ['Geoff', 'Fusion'], ['Trance', 'Former'], ['Jen', 'Aya
i']]"
}
[chain/start] [1:chain:AgentExecutor > 2:chain:LLMChain] Entering Chain ru
n with input:
{
  "input": "Sort these customers by last name and then first name and prin
t the output: [['Harrison', 'Chase'], ['Lang', 'Chain'], ['Dolly', 'Too'],
['Elle', 'Elem'], ['Geoff', 'Fusion'], ['Trance', 'Former'], ['Jen', 'Aya
i']]",
  "agent_scratchpad": "",
  "stop": [
    "\nObservation:",
    "\n\tObservation:"
  ]
}
```

Define your own tool

In [30]:
```
#!pip install DateTime
```

In [31]:
```
from langchain.agents import tool
from datetime import date
```

In [32]:
```
@tool
def time(text: str) -> str:
    """Returns todays date, use this for any \
    questions related to knowing todays date. \
    The input should always be an empty string, \
    and this function will always return todays \
    date - any date mathmatics should occur \
    outside this function."""
    return str(date.today())
```

In [33]:
```
agent= initialize_agent(
    tools + [time],
    llm,
    agent=AgentType.CHAT_ZERO_SHOT_REACT_DESCRIPTION,
    handle_parsing_errors=True,
    verbose = True)
```

Note:

The agent will sometimes come to the wrong conclusion (agents are a work in progress!).

If it does, please try running it again.

In [34]:
```
try:
    result = agent("whats the date today?")
except:
    print("exception on external access")
```

```
> Entering new AgentExecutor chain...
Thought: I need to use the `time` tool to get today's date.
Action:
```
{
  "action": "time",
  "action_input": ""
}
```

Observation: 2023-06-24
Thought:I have successfully retrieved today's date using the `time` tool.
Final Answer: Today's date is 2023-06-24.

> Finished chain.
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: