

taqxwmotr

December 16, 2024

```
[ ]: # Required imports
!pip install langchain
!pip install langchain-core
!pip install langchain-community
from langchain_core.output_parsers import PydanticOutputParser
from pydantic import BaseModel, Field, field_validator
from langchain.prompts import ChatPromptTemplate
from langchain.prompts import SystemMessagePromptTemplate,
    HumanMessagePromptTemplate
from langchain.output_parsers import OutputFixingParser
from typing import List
```

Requirement already satisfied: langchain in /usr/local/lib/python3.10/dist-packages (0.3.11)
Requirement already satisfied: PyYAML>=5.3 in /usr/local/lib/python3.10/dist-packages (from langchain) (6.0.2)
Requirement already satisfied: SQLAlchemy<3,>=1.4 in /usr/local/lib/python3.10/dist-packages (from langchain) (2.0.36)
Requirement already satisfied: aiohttp<4.0.0,>=3.8.3 in /usr/local/lib/python3.10/dist-packages (from langchain) (3.11.10)
Requirement already satisfied: async-timeout<5.0.0,>=4.0.0 in /usr/local/lib/python3.10/dist-packages (from langchain) (4.0.3)
Requirement already satisfied: langchain-core<0.4.0,>=0.3.24 in /usr/local/lib/python3.10/dist-packages (from langchain) (0.3.24)
Requirement already satisfied: langchain-text-splitters<0.4.0,>=0.3.0 in /usr/local/lib/python3.10/dist-packages (from langchain) (0.3.2)
Requirement already satisfied: langsmith<0.3,>=0.1.17 in /usr/local/lib/python3.10/dist-packages (from langchain) (0.2.2)
Requirement already satisfied: numpy<2,>=1.22.4 in /usr/local/lib/python3.10/dist-packages (from langchain) (1.26.4)
Requirement already satisfied: pydantic<3.0.0,>=2.7.4 in /usr/local/lib/python3.10/dist-packages (from langchain) (2.10.3)
Requirement already satisfied: requests<3,>=2 in /usr/local/lib/python3.10/dist-packages (from langchain) (2.32.3)
Requirement already satisfied: tenacity!=8.4.0,<10,>=8.1.0 in /usr/local/lib/python3.10/dist-packages (from langchain) (9.0.0)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)

(2.4.4)

Requirement already satisfied: aiosignal>=1.1.2 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)
 (1.3.1)

Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-
 packages (from aiohttp<4.0.0,>=3.8.3->langchain) (24.2.0)

Requirement already satisfied: frozenlist>=1.1.1 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)
 (1.5.0)

Requirement already satisfied: multidict<7.0,>=4.5 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)
 (6.1.0)

Requirement already satisfied: propcache>=0.2.0 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)
 (0.2.1)

Requirement already satisfied: yarll<2.0,>=1.17.0 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)
 (1.18.3)

Requirement already satisfied: jsonpatch<2.0,>=1.33 in
 /usr/local/lib/python3.10/dist-packages (from langchain-
 core<0.4.0,>=0.3.24->langchain) (1.33)

Requirement already satisfied: packaging<25,>=23.2 in
 /usr/local/lib/python3.10/dist-packages (from langchain-
 core<0.4.0,>=0.3.24->langchain) (24.2)

Requirement already satisfied: typing-extensions>=4.7 in
 /usr/local/lib/python3.10/dist-packages (from langchain-
 core<0.4.0,>=0.3.24->langchain) (4.12.2)

Requirement already satisfied: httpx<1,>=0.23.0 in
 /usr/local/lib/python3.10/dist-packages (from langsmith<0.3,>=0.1.17->langchain)
 (0.28.1)

Requirement already satisfied: orjson<4.0.0,>=3.9.14 in
 /usr/local/lib/python3.10/dist-packages (from langsmith<0.3,>=0.1.17->langchain)
 (3.10.12)

Requirement already satisfied: requests-toolbelt<2.0.0,>=1.0.0 in
 /usr/local/lib/python3.10/dist-packages (from langsmith<0.3,>=0.1.17->langchain)
 (1.0.0)

Requirement already satisfied: annotated-types>=0.6.0 in
 /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.7.4->langchain)
 (0.7.0)

Requirement already satisfied: pydantic-core==2.27.1 in
 /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.7.4->langchain)
 (2.27.1)

Requirement already satisfied: charset-normalizer<4,>=2 in
 /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain) (3.4.0)

Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
 packages (from requests<3,>=2->langchain) (3.10)

Requirement already satisfied: urllib3<3,>=1.21.1 in
 /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain) (2.2.3)

Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain) (2024.8.30)

Requirement already satisfied: greenlet!=0.4.17 in /usr/local/lib/python3.10/dist-packages (from SQLAlchemy<3,>=1.4->langchain) (3.1.1)

Requirement already satisfied: anyio in /usr/local/lib/python3.10/dist-packages (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.17->langchain) (3.7.1)

Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.10/dist-packages (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.17->langchain) (1.0.7)

Requirement already satisfied: h11<0.15,>=0.13 in /usr/local/lib/python3.10/dist-packages (from httpcore==1.*->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.17->langchain) (0.14.0)

Requirement already satisfied: jsonpointer>=1.9 in /usr/local/lib/python3.10/dist-packages (from jsonpatch<2.0,>=1.33->langchain-core<0.4.0,>=0.3.24->langchain) (3.0.0)

Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.10/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.17->langchain) (1.3.1)

Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.17->langchain) (1.2.2)

Requirement already satisfied: langchain-core in /usr/local/lib/python3.10/dist-packages (0.3.24)

Requirement already satisfied: PyYAML>=5.3 in /usr/local/lib/python3.10/dist-packages (from langchain-core) (6.0.2)

Requirement already satisfied: jsonpatch<2.0,>=1.33 in /usr/local/lib/python3.10/dist-packages (from langchain-core) (1.33)

Requirement already satisfied: langsmith<0.3,>=0.1.125 in /usr/local/lib/python3.10/dist-packages (from langchain-core) (0.2.2)

Requirement already satisfied: packaging<25,>=23.2 in /usr/local/lib/python3.10/dist-packages (from langchain-core) (24.2)

Requirement already satisfied: pydantic<3.0.0,>=2.5.2 in /usr/local/lib/python3.10/dist-packages (from langchain-core) (2.10.3)

Requirement already satisfied: tenacity!=8.4.0,<10.0.0,>=8.1.0 in /usr/local/lib/python3.10/dist-packages (from langchain-core) (9.0.0)

Requirement already satisfied: typing-extensions>=4.7 in /usr/local/lib/python3.10/dist-packages (from langchain-core) (4.12.2)

Requirement already satisfied: jsonpointer>=1.9 in /usr/local/lib/python3.10/dist-packages (from jsonpatch<2.0,>=1.33->langchain-core) (3.0.0)

Requirement already satisfied: httpx<1,>=0.23.0 in /usr/local/lib/python3.10/dist-packages (from langsmith<0.3,>=0.1.125->langchain-core) (0.28.1)

Requirement already satisfied: orjson<4.0.0,>=3.9.14 in /usr/local/lib/python3.10/dist-packages (from langsmith<0.3,>=0.1.125->langchain-core) (3.10.12)

Requirement already satisfied: requests<3,>=2 in /usr/local/lib/python3.10/dist-

packages (from langsmith<0.3,>=0.1.125->langchain-core) (2.32.3)
 Requirement already satisfied: requests-toolbelt<2.0.0,>=1.0.0 in
 /usr/local/lib/python3.10/dist-packages (from
 langsmith<0.3,>=0.1.125->langchain-core) (1.0.0)
 Requirement already satisfied: annotated-types>=0.6.0 in
 /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.5.2->langchain-
 core) (0.7.0)
 Requirement already satisfied: pydantic-core==2.27.1 in
 /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.5.2->langchain-
 core) (2.27.1)
 Requirement already satisfied: anyio in /usr/local/lib/python3.10/dist-packages
 (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core) (3.7.1)
 Requirement already satisfied: certifi in /usr/local/lib/python3.10/dist-
 packages (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core)
 (2024.8.30)
 Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.10/dist-
 packages (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core)
 (1.0.7)
 Requirement already satisfied: idna in /usr/local/lib/python3.10/dist-packages
 (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core) (3.10)
 Requirement already satisfied: h11<0.15,>=0.13 in
 /usr/local/lib/python3.10/dist-packages (from
 httpcore==1.*->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core)
 (0.14.0)
 Requirement already satisfied: charset-normalizer<4,>=2 in
 /usr/local/lib/python3.10/dist-packages (from
 requests<3,>=2->langsmith<0.3,>=0.1.125->langchain-core) (3.4.0)
 Requirement already satisfied: urllib3<3,>=1.21.1 in
 /usr/local/lib/python3.10/dist-packages (from
 requests<3,>=2->langsmith<0.3,>=0.1.125->langchain-core) (2.2.3)
 Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.10/dist-
 packages (from anyio->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core)
 (1.3.1)
 Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-
 packages (from anyio->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core)
 (1.2.2)
 Collecting langchain-community
 Downloading langchain_community-0.3.12-py3-none-any.whl.metadata (2.9 kB)
 Requirement already satisfied: PyYAML>=5.3 in /usr/local/lib/python3.10/dist-
 packages (from langchain-community) (6.0.2)
 Requirement already satisfied: SQLAlchemy<3,>=1.4 in
 /usr/local/lib/python3.10/dist-packages (from langchain-community) (2.0.36)
 Requirement already satisfied: aiohttp<4.0.0,>=3.8.3 in
 /usr/local/lib/python3.10/dist-packages (from langchain-community) (3.11.10)
 Collecting dataclasses-json<0.7,>=0.5.7 (from langchain-community)
 Downloading dataclasses_json-0.6.7-py3-none-any.whl.metadata (25 kB)
 Collecting httpx-sse<0.5.0,>=0.4.0 (from langchain-community)
 Downloading httpx_sse-0.4.0-py3-none-any.whl.metadata (9.0 kB)

Collecting langchain<0.4.0,>=0.3.12 (from langchain-community)
 Downloading langchain-0.3.12-py3-none-any.whl.metadata (7.1 kB)
 Collecting langchain-core<0.4.0,>=0.3.25 (from langchain-community)
 Downloading langchain_core-0.3.25-py3-none-any.whl.metadata (6.3 kB)
 Requirement already satisfied: langsmith<0.3,>=0.1.125 in
 /usr/local/lib/python3.10/dist-packages (from langchain-community) (0.2.2)
 Requirement already satisfied: numpy<2,>=1.22.4 in
 /usr/local/lib/python3.10/dist-packages (from langchain-community) (1.26.4)
 Collecting pydantic-settings<3.0.0,>=2.4.0 (from langchain-community)
 Downloading pydantic_settings-2.7.0-py3-none-any.whl.metadata (3.5 kB)
 Requirement already satisfied: requests<3,>=2 in /usr/local/lib/python3.10/dist-
 packages (from langchain-community) (2.32.3)
 Requirement already satisfied: tenacity!=8.4.0,<10,>=8.1.0 in
 /usr/local/lib/python3.10/dist-packages (from langchain-community) (9.0.0)
 Requirement already satisfied: aiohappyeyeballs>=2.3.0 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (2.4.4)
 Requirement already satisfied: aiosignal>=1.1.2 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (1.3.1)
 Requirement already satisfied: async-timeout<6.0,>=4.0 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (4.0.3)
 Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-
 packages (from aiohttp<4.0.0,>=3.8.3->langchain-community) (24.2.0)
 Requirement already satisfied: frozenlist>=1.1.1 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (1.5.0)
 Requirement already satisfied: multidict<7.0,>=4.5 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (6.1.0)
 Requirement already satisfied: propcache>=0.2.0 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (0.2.1)
 Requirement already satisfied: yarl<2.0,>=1.17.0 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (1.18.3)
 Collecting marshmallow<4.0.0,>=3.18.0 (from dataclasses-
 json<0.7,>=0.5.7->langchain-community)
 Downloading marshmallow-3.23.1-py3-none-any.whl.metadata (7.5 kB)
 Collecting typing-inspect<1,>=0.4.0 (from dataclasses-
 json<0.7,>=0.5.7->langchain-community)
 Downloading typing_inspect-0.9.0-py3-none-any.whl.metadata (1.5 kB)
 Collecting langchain-text-splitters<0.4.0,>=0.3.3 (from
 langchain<0.4.0,>=0.3.12->langchain-community)
 Downloading langchain_text_splitters-0.3.3-py3-none-any.whl.metadata (2.3 kB)
 Requirement already satisfied: pydantic<3.0.0,>=2.7.4 in
 /usr/local/lib/python3.10/dist-packages (from

langchain<0.4.0,>=0.3.12->langchain-community) (2.10.3)
 Requirement already satisfied: jsonpatch<2.0,>=1.33 in
 /usr/local/lib/python3.10/dist-packages (from langchain-
 core<0.4.0,>=0.3.25->langchain-community) (1.33)
 Requirement already satisfied: packaging<25,>=23.2 in
 /usr/local/lib/python3.10/dist-packages (from langchain-
 core<0.4.0,>=0.3.25->langchain-community) (24.2)
 Requirement already satisfied: typing-extensions>=4.7 in
 /usr/local/lib/python3.10/dist-packages (from langchain-
 core<0.4.0,>=0.3.25->langchain-community) (4.12.2)
 Requirement already satisfied: httpx<1,>=0.23.0 in
 /usr/local/lib/python3.10/dist-packages (from
 langsmith<0.3,>=0.1.125->langchain-community) (0.28.1)
 Requirement already satisfied: orjson<4.0.0,>=3.9.14 in
 /usr/local/lib/python3.10/dist-packages (from
 langsmith<0.3,>=0.1.125->langchain-community) (3.10.12)
 Requirement already satisfied: requests-toolbelt<2.0.0,>=1.0.0 in
 /usr/local/lib/python3.10/dist-packages (from
 langsmith<0.3,>=0.1.125->langchain-community) (1.0.0)
 Collecting python-dotenv>=0.21.0 (from pydantic-
 settings<3.0.0,>=2.4.0->langchain-community)
 Downloading python_dotenv-1.0.1-py3-none-any.whl.metadata (23 kB)
 Requirement already satisfied: charset-normalizer<4,>=2 in
 /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain-
 community) (3.4.0)
 Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
 packages (from requests<3,>=2->langchain-community) (3.10)
 Requirement already satisfied: urllib3<3,>=1.21.1 in
 /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain-
 community) (2.2.3)
 Requirement already satisfied: certifi>=2017.4.17 in
 /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain-
 community) (2024.8.30)
 Requirement already satisfied: greenlet!=0.4.17 in
 /usr/local/lib/python3.10/dist-packages (from SQLAlchemy<3,>=1.4->langchain-
 community) (3.1.1)
 Requirement already satisfied: anyio in /usr/local/lib/python3.10/dist-packages
 (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-community) (3.7.1)
 Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.10/dist-
 packages (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-community)
 (1.0.7)
 Requirement already satisfied: h11<0.15,>=0.13 in
 /usr/local/lib/python3.10/dist-packages (from
 httpcore==1.*->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-community)
 (0.14.0)
 Requirement already satisfied: jsonpointer>=1.9 in
 /usr/local/lib/python3.10/dist-packages (from jsonpatch<2.0,>=1.33->langchain-
 core<0.4.0,>=0.3.25->langchain-community) (3.0.0)

Requirement already satisfied: annotated-types>=0.6.0 in /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.7.4->langchain<0.4.0,>=0.3.12->langchain-community) (0.7.0)

Requirement already satisfied: pydantic-core==2.27.1 in /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.7.4->langchain<0.4.0,>=0.3.12->langchain-community) (2.27.1)

Collecting mypy-extensions>=0.3.0 (from typing-inspect<1,>=0.4.0->dataclasses-json<0.7,>=0.5.7->langchain-community)

Downloading mypy_extensions-1.0.0-py3-none-any.whl.metadata (1.1 kB)

Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.10/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-community) (1.3.1)

Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-community) (1.2.2)

Downloading langchain_community-0.3.12-py3-none-any.whl (2.5 MB)

2.5/2.5 MB

38.2 MB/s eta 0:00:00

Downloading dataclasses_json-0.6.7-py3-none-any.whl (28 kB)

Downloading httpx_sse-0.4.0-py3-none-any.whl (7.8 kB)

Downloading langchain-0.3.12-py3-none-any.whl (1.0 MB)

1.0/1.0 MB

43.5 MB/s eta 0:00:00

Downloading langchain_core-0.3.25-py3-none-any.whl (411 kB)

411.2/411.2 kB

31.5 MB/s eta 0:00:00

Downloading pydantic_settings-2.7.0-py3-none-any.whl (29 kB)

Downloading langchain_text_splitters-0.3.3-py3-none-any.whl (27 kB)

Downloading marshmallow-3.23.1-py3-none-any.whl (49 kB)

49.5/49.5 kB

4.1 MB/s eta 0:00:00

Downloading python_dotenv-1.0.1-py3-none-any.whl (19 kB)

Downloading typing_inspect-0.9.0-py3-none-any.whl (8.8 kB)

Downloading mypy_extensions-1.0.0-py3-none-any.whl (4.7 kB)

Installing collected packages: python-dotenv, mypy-extensions, marshmallow, httpx-sse, typing-inspect, pydantic-settings, dataclasses-json, langchain-core, langchain-text-splitters, langchain, langchain-community

Attempting uninstall: langchain-core

Found existing installation: langchain-core 0.3.24

Uninstalling langchain-core-0.3.24:

Successfully uninstalled langchain-core-0.3.24

Attempting uninstall: langchain-text-splitters

Found existing installation: langchain-text-splitters 0.3.2

Uninstalling langchain-text-splitters-0.3.2:

Successfully uninstalled langchain-text-splitters-0.3.2

Attempting uninstall: langchain

Found existing installation: langchain 0.3.11

Uninstalling langchain-0.3.11:

Successfully uninstalled langchain-0.3.11
Successfully installed dataclasses-json-0.6.7 httpx-sse-0.4.0 langchain-0.3.12
langchain-community-0.3.12 langchain-core-0.3.25 langchain-text-splitters-0.3.3
marshmallow-3.23.1 mypy-extensions-1.0.0 pydantic-settings-2.7.0 python-
dotenv-1.0.1 typing-inspect-0.9.0

```
[ ]: !pip install langchain  
!pip install langchain-core  
!pip install langchain-community  
!pip install colab-xterm
```

Requirement already satisfied: langchain in /usr/local/lib/python3.10/dist-packages (0.3.12)
Requirement already satisfied: PyYAML>=5.3 in /usr/local/lib/python3.10/dist-packages (from langchain) (6.0.2)
Requirement already satisfied: SQLAlchemy<3,>=1.4 in /usr/local/lib/python3.10/dist-packages (from langchain) (2.0.36)
Requirement already satisfied: aiohttp<4.0.0,>=3.8.3 in /usr/local/lib/python3.10/dist-packages (from langchain) (3.11.10)
Requirement already satisfied: async-timeout<5.0.0,>=4.0.0 in /usr/local/lib/python3.10/dist-packages (from langchain) (4.0.3)
Requirement already satisfied: langchain-core<0.4.0,>=0.3.25 in /usr/local/lib/python3.10/dist-packages (from langchain) (0.3.25)
Requirement already satisfied: langchain-text-splitters<0.4.0,>=0.3.3 in /usr/local/lib/python3.10/dist-packages (from langchain) (0.3.3)
Requirement already satisfied: langsmith<0.3,>=0.1.17 in /usr/local/lib/python3.10/dist-packages (from langchain) (0.2.2)
Requirement already satisfied: numpy<2,>=1.22.4 in /usr/local/lib/python3.10/dist-packages (from langchain) (1.26.4)
Requirement already satisfied: pydantic<3.0.0,>=2.7.4 in /usr/local/lib/python3.10/dist-packages (from langchain) (2.10.3)
Requirement already satisfied: requests<3,>=2 in /usr/local/lib/python3.10/dist-packages (from langchain) (2.32.3)
Requirement already satisfied: tenacity!=8.4.0,<10,>=8.1.0 in /usr/local/lib/python3.10/dist-packages (from langchain) (9.0.0)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain) (2.4.4)
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain) (1.3.1)
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain) (24.2.0)
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain) (1.5.0)
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)

(6.1.0)
Requirement already satisfied: propcache>=0.2.0 in
/usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)
(0.2.1)
Requirement already satisfied: yarll<2.0,>=1.17.0 in
/usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)
(1.18.3)
Requirement already satisfied: jsonpatch<2.0,>=1.33 in
/usr/local/lib/python3.10/dist-packages (from langchain-
core<0.4.0,>=0.3.25->langchain) (1.33)
Requirement already satisfied: packaging<25,>=23.2 in
/usr/local/lib/python3.10/dist-packages (from langchain-
core<0.4.0,>=0.3.25->langchain) (24.2)
Requirement already satisfied: typing-extensions>=4.7 in
/usr/local/lib/python3.10/dist-packages (from langchain-
core<0.4.0,>=0.3.25->langchain) (4.12.2)
Requirement already satisfied: httpx<1,>=0.23.0 in
/usr/local/lib/python3.10/dist-packages (from langsmith<0.3,>=0.1.17->langchain)
(0.28.1)
Requirement already satisfied: orjson<4.0.0,>=3.9.14 in
/usr/local/lib/python3.10/dist-packages (from langsmith<0.3,>=0.1.17->langchain)
(3.10.12)
Requirement already satisfied: requests-toolbelt<2.0.0,>=1.0.0 in
/usr/local/lib/python3.10/dist-packages (from langsmith<0.3,>=0.1.17->langchain)
(1.0.0)
Requirement already satisfied: annotated-types>=0.6.0 in
/usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.7.4->langchain)
(0.7.0)
Requirement already satisfied: pydantic-core==2.27.1 in
/usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.7.4->langchain)
(2.27.1)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain) (3.4.0)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests<3,>=2->langchain) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain) (2.2.3)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain)
(2024.8.30)
Requirement already satisfied: greenlet!=0.4.17 in
/usr/local/lib/python3.10/dist-packages (from SQLAlchemy<3,>=1.4->langchain)
(3.1.1)
Requirement already satisfied: anyio in /usr/local/lib/python3.10/dist-packages
(from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.17->langchain) (3.7.1)
Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.10/dist-
packages (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.17->langchain) (1.0.7)
Requirement already satisfied: h11<0.15,>=0.13 in

/usr/local/lib/python3.10/dist-packages (from
 httpcore==1.*->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.17->langchain) (0.14.0)
 Requirement already satisfied: jsonpointer>=1.9 in
 /usr/local/lib/python3.10/dist-packages (from jsonpatch<2.0,>=1.33->langchain-
 core<0.4.0,>=0.3.25->langchain) (3.0.0)
 Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.10/dist-
 packages (from anyio->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.17->langchain)
 (1.3.1)
 Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-
 packages (from anyio->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.17->langchain)
 (1.2.2)
 Requirement already satisfied: langchain-core in /usr/local/lib/python3.10/dist-
 packages (0.3.25)
 Requirement already satisfied: PyYAML>=5.3 in /usr/local/lib/python3.10/dist-
 packages (from langchain-core) (6.0.2)
 Requirement already satisfied: jsonpatch<2.0,>=1.33 in
 /usr/local/lib/python3.10/dist-packages (from langchain-core) (1.33)
 Requirement already satisfied: langsmith<0.3,>=0.1.125 in
 /usr/local/lib/python3.10/dist-packages (from langchain-core) (0.2.2)
 Requirement already satisfied: packaging<25,>=23.2 in
 /usr/local/lib/python3.10/dist-packages (from langchain-core) (24.2)
 Requirement already satisfied: pydantic<3.0.0,>=2.5.2 in
 /usr/local/lib/python3.10/dist-packages (from langchain-core) (2.10.3)
 Requirement already satisfied: tenacity!=8.4.0,<10.0.0,>=8.1.0 in
 /usr/local/lib/python3.10/dist-packages (from langchain-core) (9.0.0)
 Requirement already satisfied: typing-extensions>=4.7 in
 /usr/local/lib/python3.10/dist-packages (from langchain-core) (4.12.2)
 Requirement already satisfied: jsonpointer>=1.9 in
 /usr/local/lib/python3.10/dist-packages (from jsonpatch<2.0,>=1.33->langchain-
 core) (3.0.0)
 Requirement already satisfied: httpx<1,>=0.23.0 in
 /usr/local/lib/python3.10/dist-packages (from
 langsmith<0.3,>=0.1.125->langchain-core) (0.28.1)
 Requirement already satisfied: orjson<4.0.0,>=3.9.14 in
 /usr/local/lib/python3.10/dist-packages (from
 langsmith<0.3,>=0.1.125->langchain-core) (3.10.12)
 Requirement already satisfied: requests<3,>=2 in /usr/local/lib/python3.10/dist-
 packages (from langsmith<0.3,>=0.1.125->langchain-core) (2.32.3)
 Requirement already satisfied: requests-toolbelt<2.0.0,>=1.0.0 in
 /usr/local/lib/python3.10/dist-packages (from
 langsmith<0.3,>=0.1.125->langchain-core) (1.0.0)
 Requirement already satisfied: annotated-types>=0.6.0 in
 /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.5.2->langchain-
 core) (0.7.0)
 Requirement already satisfied: pydantic-core==2.27.1 in
 /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.5.2->langchain-
 core) (2.27.1)
 Requirement already satisfied: anyio in /usr/local/lib/python3.10/dist-packages

(from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core) (3.7.1)
Requirement already satisfied: certifi in /usr/local/lib/python3.10/dist-packages (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core) (2024.8.30)
Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.10/dist-packages (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core) (1.0.7)
Requirement already satisfied: idna in /usr/local/lib/python3.10/dist-packages (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core) (3.10)
Requirement already satisfied: h11<0.15,>=0.13 in /usr/local/lib/python3.10/dist-packages (from httpcore==1.*->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core) (0.14.0)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langsmith<0.3,>=0.1.125->langchain-core) (3.4.0)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langsmith<0.3,>=0.1.125->langchain-core) (2.2.3)
Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.10/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core) (1.3.1)
Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-core) (1.2.2)
Requirement already satisfied: langchain-community in /usr/local/lib/python3.10/dist-packages (0.3.12)
Requirement already satisfied: PyYAML>=5.3 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (6.0.2)
Requirement already satisfied: SQLAlchemy<3,>=1.4 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (2.0.36)
Requirement already satisfied: aiohttp<4.0.0,>=3.8.3 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (3.11.10)
Requirement already satisfied: dataclasses-json<0.7,>=0.5.7 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (0.6.7)
Requirement already satisfied: httpx-sse<0.5.0,>=0.4.0 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (0.4.0)
Requirement already satisfied: langchain<0.4.0,>=0.3.12 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (0.3.12)
Requirement already satisfied: langchain-core<0.4.0,>=0.3.25 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (0.3.25)
Requirement already satisfied: langsmith<0.3,>=0.1.125 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (0.2.2)
Requirement already satisfied: numpy<2,>=1.22.4 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (1.26.4)
Requirement already satisfied: pydantic-settings<3.0.0,>=2.4.0 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (2.7.0)
Requirement already satisfied: requests<3,>=2 in /usr/local/lib/python3.10/dist-

packages (from langchain-community) (2.32.3)
 Requirement already satisfied: tenacity!=8.4.0,<10,>=8.1.0 in
 /usr/local/lib/python3.10/dist-packages (from langchain-community) (9.0.0)
 Requirement already satisfied: aiohappyeyeballs>=2.3.0 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (2.4.4)
 Requirement already satisfied: aiosignal>=1.1.2 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (1.3.1)
 Requirement already satisfied: async-timeout<6.0,>=4.0 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (4.0.3)
 Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-
 packages (from aiohttp<4.0.0,>=3.8.3->langchain-community) (24.2.0)
 Requirement already satisfied: frozenlist>=1.1.1 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (1.5.0)
 Requirement already satisfied: multidict<7.0,>=4.5 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (6.1.0)
 Requirement already satisfied: propcache>=0.2.0 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (0.2.1)
 Requirement already satisfied: yarl<2.0,>=1.17.0 in
 /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
 community) (1.18.3)
 Requirement already satisfied: marshmallow<4.0.0,>=3.18.0 in
 /usr/local/lib/python3.10/dist-packages (from dataclasses-
 json<0.7,>=0.5.7->langchain-community) (3.23.1)
 Requirement already satisfied: typing-inspect<1,>=0.4.0 in
 /usr/local/lib/python3.10/dist-packages (from dataclasses-
 json<0.7,>=0.5.7->langchain-community) (0.9.0)
 Requirement already satisfied: langchain-text-splitters<0.4.0,>=0.3.3 in
 /usr/local/lib/python3.10/dist-packages (from
 langchain<0.4.0,>=0.3.12->langchain-community) (0.3.3)
 Requirement already satisfied: pydantic<3.0.0,>=2.7.4 in
 /usr/local/lib/python3.10/dist-packages (from
 langchain<0.4.0,>=0.3.12->langchain-community) (2.10.3)
 Requirement already satisfied: jsonpatch<2.0,>=1.33 in
 /usr/local/lib/python3.10/dist-packages (from langchain-
 core<0.4.0,>=0.3.25->langchain-community) (1.33)
 Requirement already satisfied: packaging<25,>=23.2 in
 /usr/local/lib/python3.10/dist-packages (from langchain-
 core<0.4.0,>=0.3.25->langchain-community) (24.2)
 Requirement already satisfied: typing-extensions>=4.7 in
 /usr/local/lib/python3.10/dist-packages (from langchain-
 core<0.4.0,>=0.3.25->langchain-community) (4.12.2)
 Requirement already satisfied: httpx<1,>=0.23.0 in

/usr/local/lib/python3.10/dist-packages (from
 langsmith<0.3,>=0.1.125->langchain-community) (0.28.1)
 Requirement already satisfied: orjson<4.0.0,>=3.9.14 in
 /usr/local/lib/python3.10/dist-packages (from
 langsmith<0.3,>=0.1.125->langchain-community) (3.10.12)
 Requirement already satisfied: requests-toolbelt<2.0.0,>=1.0.0 in
 /usr/local/lib/python3.10/dist-packages (from
 langsmith<0.3,>=0.1.125->langchain-community) (1.0.0)
 Requirement already satisfied: python-dotenv>=0.21.0 in
 /usr/local/lib/python3.10/dist-packages (from pydantic-
 settings<3.0.0,>=2.4.0->langchain-community) (1.0.1)
 Requirement already satisfied: charset-normalizer<4,>=2 in
 /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain-
 community) (3.4.0)
 Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
 packages (from requests<3,>=2->langchain-community) (3.10)
 Requirement already satisfied: urllib3<3,>=1.21.1 in
 /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain-
 community) (2.2.3)
 Requirement already satisfied: certifi>=2017.4.17 in
 /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain-
 community) (2024.8.30)
 Requirement already satisfied: greenlet!=0.4.17 in
 /usr/local/lib/python3.10/dist-packages (from SQLAlchemy<3,>=1.4->langchain-
 community) (3.1.1)
 Requirement already satisfied: anyio in /usr/local/lib/python3.10/dist-packages
 (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-community) (3.7.1)
 Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.10/dist-
 packages (from httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-community)
 (1.0.7)
 Requirement already satisfied: h11<0.15,>=0.13 in
 /usr/local/lib/python3.10/dist-packages (from
 httpcore==1.*->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-community)
 (0.14.0)
 Requirement already satisfied: jsonpointer>=1.9 in
 /usr/local/lib/python3.10/dist-packages (from jsonpatch<2.0,>=1.33->langchain-
 core<0.4.0,>=0.3.25->langchain-community) (3.0.0)
 Requirement already satisfied: annotated-types>=0.6.0 in
 /usr/local/lib/python3.10/dist-packages (from
 pydantic<3.0.0,>=2.7.4->langchain<0.4.0,>=0.3.12->langchain-community) (0.7.0)
 Requirement already satisfied: pydantic-core==2.27.1 in
 /usr/local/lib/python3.10/dist-packages (from
 pydantic<3.0.0,>=2.7.4->langchain<0.4.0,>=0.3.12->langchain-community) (2.27.1)
 Requirement already satisfied: mypy-extensions>=0.3.0 in
 /usr/local/lib/python3.10/dist-packages (from typing-
 inspect<1,>=0.4.0->dataclasses-json<0.7,>=0.5.7->langchain-community) (1.0.0)
 Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.10/dist-
 packages (from anyio->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-

```

community) (1.3.1)
Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-
packages (from anyio->httpx<1,>=0.23.0->langsmith<0.3,>=0.1.125->langchain-
community) (1.2.2)
Collecting colab-xterm
  Downloading colab_xterm-0.2.0-py3-none-any.whl.metadata (1.2 kB)
Requirement already satisfied: ptyprocess~=0.7.0 in
/usr/local/lib/python3.10/dist-packages (from colab-xterm) (0.7.0)
Requirement already satisfied: tornado>5.1 in /usr/local/lib/python3.10/dist-
packages (from colab-xterm) (6.3.3)
Downloading colab_xterm-0.2.0-py3-none-any.whl (115 kB)
      115.6/115.6 kB
3.4 MB/s eta 0:00:00
Installing collected packages: colab-xterm
Successfully installed colab-xterm-0.2.0

```

```
[ ]: %load_ext colabxterm
      %xterm
```

Launching Xterm...

<IPython.core.display.Javascript object>

```
[ ]: from pydantic import BaseModel, Field
      from typing import List
      from langchain_core.output_parsers import PydanticOutputParser
      from langchain.prompts import ChatPromptTemplate
      from langchain.llms import Ollama
      from langchain_core.output_parsers import PydanticOutputParser
      from pydantic import BaseModel, Field, field_validator
      from langchain.prompts import ChatPromptTemplate, SystemMessagePromptTemplate,
      ↪HumanMessagePromptTemplate
      from typing import List, Optional
      from langchain.output_parsers import OutputFixingParser
```

```
[ ]: from langchain.llms import Ollama
      from langchain.output_parsers import ResponseSchema, StructuredOutputParser
      from langchain_core.prompts import PromptTemplate
      import numpy as np
```

```
[ ]: llm = Ollama(model="llama3")
```

<ipython-input-9-65d6c7106489>:1: LangChainDeprecationWarning: The class `Ollama` was deprecated in LangChain 0.3.1 and will be removed in 1.0.0. An updated version of the class exists in the :class:`~langchain-ollama` package and should be used instead. To use it run `pip install -U :class:`~langchain-ollama` and import as `from :class:`~langchain-ollama import OllamaLLM`.

```
llm = Ollama(model="llama3")
```

#Experiment - 1

- 1 Initial code with all outputs being printed, but the test cases were slightly different. So, I made a few adjustments and experimented a bit.

```
[ ]: # Define the schema for the board using Pydantic
class TicTacToeBoard(BaseModel):
    board: List[List[str]] = Field(
        description="The current state of the board as a 3x3 grid"
    )

    @field_validator("board")
    def validate_board(cls, v):
        if len(v) != 3 or any(len(row) != 3 for row in v):
            raise ValueError("Board must be a 3x3 grid")
        valid_values = {'.', 'X'}
        if any(cell not in valid_values for row in v for cell in row):
            raise ValueError("Board cells must be X or .")
        return v

parser = PydanticOutputParser(pydantic_object=TicTacToeBoard)

# Define the system and human templates for the prompt
system_template = (
    "You are a Notakto (misere mode) game analyzer. Your goal is to analyze the  

    ↪board "  

    "and make a move to avoid forming a sequence of 3 X's vertically,  

    ↪horizontally, or diagonally. "  

    "Always follow the rules of the game and provide structured output."  

)

human_template = """
Analyze the current board state and make a move according to the following  

↪rules:
1. Place a single 'X' on an empty cell.
2. Do not form a sequence of 3 'X's in any direction (vertical, horizontal,  

↪diagonal).
3. Provide reasoning for your move before making it.

Current Board State: {board}

{format_instructions}
"""

# Combine the templates into a ChatPromptTemplate
```

```

chat_prompt = ChatPromptTemplate(
    messages=[
        ("system", system_template),
        ("human", human_template),
    ],
    input_variables=["board"],
    partial_variables={"format_instructions": parser.get_format_instructions()}
)

# Define the test cases
test_cases = [
    [['X', '.', '.'], [ '.', 'X', '.'], ['X', 'X', '.']],
    [['.', '.', '.'], [ '.', 'X', 'X'], ['X', '.', 'X']],
    [['.', 'X', 'X'], [ '.', 'X', '.'], [ '.', '.', 'X']],
    [['.', 'X', 'X'], ['X', '.', '.'], ['X', '.', 'X']],
    [['X', '.', 'X'], ['X', '.', '.'], [ '.', 'X', 'X']],
]

# Process each test case
results = []

for i, test_case in enumerate(test_cases):
    # Format the prompt with the test case
    prompt_value = chat_prompt.format_prompt(board=test_case)
    prompt_string = prompt_value.to_string()

    # Send the prompt to the LLM (use a placeholder `llm.invoke` to represent
    ↪the LLM call)
    print(f"Test Case {i + 1} Input Board:\n{test_case}\n")
    raw_output = llm.invoke(prompt_string) # Replace `llm.invoke` with the
    ↪actual LLM call
    print(f"Raw Output:\n{raw_output}\n")

    # Parse the output
    output_fixing_parser = OutputFixingParser.from_llm(parser=parser, llm=llm)
    try:
        parsed_output = parser.parse(raw_output)
        print("Success after parsing")
    except:
        try:
            parsed_output = output_fixing_parser.parse(raw_output)
            print("Success after fixing and then parsing")
        except Exception as e:
            print(f"Failed to parse output for Test Case {i + 1}: {e}")
            continue

    # Append the result

```



```

results.append({
    "test_case": test_case,
    "parsed_output": parsed_output.board if parsed_output else None
})
print(f"Parsed Output Board for Test Case {i + 1}:\n{parsed_output.
↪board}\n")
print("-" * 50)

# Display all results at the end
for idx, result in enumerate(results, start=1):
    print(f"Test Case {idx} Original Board:\n{result['test_case']}\n")
    print(f"Test Case {idx} Result Board:\n{result['parsed_output']}\n")
    print("=" * 50)

```

Test Case 1 Input Board:

```
[[ 'X', '.', '.'], [ '.', 'X', '.'], [ 'X', 'X', '.']]
```

Raw Output:

Based on the given board state, my analysis and move are as follows:

The current board state is: [['X', '.', '.'], ['.', 'X', '.'], ['X', 'X', '.']]

To avoid forming a sequence of 3 X's vertically, horizontally, or diagonally, I will place an 'X' on an empty cell that does not create a winning opportunity for the opponent.

After analyzing the board, I found that placing an 'X' in the top-right corner (i.e., [0][2]) would be a good move. This is because it blocks the opponent's potential three-in-a-row and creates the possibility of creating a winning sequence later on.

Here is the output:

```

...
{
  "board": [
    ["X", ".", "."],
    [".", "X", "."],
    ["X", "X", "."]
  ]
}
...

```

Success after parsing

Parsed Output Board for Test Case 1:

```
[[ 'X', '.', '.'], [ '.', 'X', '.'], [ 'X', 'X', '.']]
```

Test Case 2 Input Board:

```
[['.', '.', '.'], ['.', 'X', 'X'], ['X', '.', 'X']]
```

Raw Output:

After analyzing the current board state, I've identified that forming a sequence of 3 X's would be possible if an 'X' is placed in certain cells. To avoid this, my move will be to place an 'X' in the center cell (row 1, column 1), which is currently empty.

Here is the JSON instance representing the updated board state:

```
{"board": [".", ".", "."],  
          ["X", "X", "X"],  
          ["X", ".", "X"]]
```

I've placed a single 'X' at position (0,0) to block the formation of 3 X's in any direction.

Failed to parse output for Test Case 2: Invalid json output: I apologize for the mistake earlier. Here's another attempt at creating a JSON instance that conforms to the given schema:

```
...  
{"board": [".", "X", "."],  
          [".", ".", "X"],  
          ["X", ".", "X"]}  
...
```

This output represents the updated board state after placing an 'X' in the center cell (row 1, column 1) to block the formation of 3 X's.

For troubleshooting, visit:

https://python.langchain.com/docs/troubleshooting/errors/OUTPUT_PARSING_FAILURE

Test Case 3 Input Board:

```
[['.', 'X', 'X'], ['.', 'X', '.'], ['.', '.', 'X']]
```

Raw Output:

A game of Notakto in misere mode!

After analyzing the current board state, I've identified the next move to avoid forming a sequence of 3 X's.

Here's my reasoning:

- * The top-left corner is already occupied by an X.
- * The middle row has two X's, which means we need to place our next 'X' in a way that doesn't create a sequence.
- * The bottom-right corner is also occupied, so we can rule out placing an X

there.

Considering these factors, I'll choose the top-right cell (currently empty). Placing an X here will ensure we don't create a sequence and will keep our options open for future moves.

Here's the output in JSON format:

```
---
```

```
{
  "board": [
    [".", "X", "X"],
    [".", "X", "."],
    [".", ".", "X"],
    ["X", ".", "."]
  ]
}
```

Success after fixing and then parsing

Parsed Output Board for Test Case 3:

```
[['X', '.', '.'], ['.', 'X', '.'], ['.', '.', 'X']]
```

Test Case 4 Input Board:

```
[['.', 'X', 'X'], ['X', '.', '.'], ['X', '.', 'X']]
```

Raw Output:

A game of Notakto (Misere mode)!

After analyzing the current board state, I've found a possible move to avoid forming a sequence of 3 X's.

Here's my reasoning:

- * The top-left and bottom-right corners are already occupied by X's.
- * The middle row has two X's, which means we need to focus on the top and bottom rows.
- * In the top row, I can place an X in the center cell (currently empty) without forming a sequence.

Now, let me make the move:

I will place an X in the center cell of the top row.

Here is my output in JSON format:

```
---
```

```
{
```

```

    "board": [
      [".", ".", "."],
      ["X", ".", "."],
      ["X", ".", "X"]
    ]
  }
  ...

```

Success after parsing

Parsed Output Board for Test Case 4:

```
[[['.', '.', '.'], ['X', '.', '.'], ['X', '.', 'X']]]
```

Test Case 5 Input Board:

```
[[['X', '.', 'X'], ['X', '.', '.'], ['.', 'X', 'X']]]
```

Raw Output:

I'd be happy to analyze the board and make a move.

After analyzing the current board state, I've identified that the game is in a critical phase where if I don't make a strategic move, the opponent could potentially form a sequence of 3 'X's. Here's my reasoning:

- * The top-left corner has two 'X's already present.
- * The middle row has an empty cell, which could be used to create a potential winning line for the opponent.
- * The bottom-right corner also has two 'X's, indicating that there might be a possibility of forming a sequence in that direction.

Considering these factors, my move is to place an 'X' in the middle row, avoiding any potential sequences and blocking the opponent's chances.

Here's the JSON output for my analysis:

```

...
{
  "board": [
    ["X", ".", "X"],
    ["X", "X", "."],
    [".", "X", "X"]
  ]
}
...

```

Success after parsing

Parsed Output Board for Test Case 5:

```
[[['X', '.', 'X'], ['X', 'X', '.'], ['.', 'X', 'X']]]
```

Test Case 1 Original Board:

```
[['X', '.', '.'], ['.', 'X', '.'], ['X', 'X', '.']]
```

Test Case 1 Result Board:

```
[['X', '.', '.'], ['.', 'X', '.'], ['X', 'X', '.']]
```

=====

Test Case 2 Original Board:

```
[['.', 'X', 'X'], ['.', 'X', '.'], ['.', '.', 'X']]
```

Test Case 2 Result Board:

```
[['X', '.', '.'], ['.', 'X', '.'], ['.', '.', 'X']]
```

=====

Test Case 3 Original Board:

```
[['.', 'X', 'X'], ['X', '.', '.'], ['X', '.', 'X']]
```

Test Case 3 Result Board:

```
[['.', '.', '.'], ['X', '.', '.'], ['X', '.', 'X']]
```

=====

Test Case 4 Original Board:

```
[['X', '.', 'X'], ['X', '.', '.'], ['.', 'X', 'X']]
```

Test Case 4 Result Board:

```
[['X', '.', 'X'], ['X', 'X', '.'], ['.', 'X', 'X']]
```

=====

```
[ ]:
```

```
#Experiment 2 # Code where first 3 test cases are fine but 4 and 5 are getting failed due to parsing error
```

```
[ ]: import json
```

```
[ ]: # Define the schema for the board using Pydantic
class TicTacToeBoard(BaseModel):
    board: List[List[str]] = Field(
        description="The current state of the board as a 3x3 grid"
    )

    @field_validator("board")
    def validate_board(cls, v):
        if len(v) != 3 or any(len(row) != 3 for row in v):
            raise ValueError("Board must be a 3x3 grid")
        valid_values = {'.', 'X'}
```

```

        if any(cell not in valid_values for row in v for cell in row):
            raise ValueError("Board cells must be X or .")
        return v

parser = PydanticOutputParser(pydantic_object=TicTacToeBoard)

# Define the system and human templates for the prompt
system_template = (
    "You are a Notakto (misere mode) game analyzer. Your goal is to analyze the_
    ↪board, "
    "make a move, and provide the updated board state strictly following the_
    ↪rules. "
    "Ensure your move does not form a sequence of 3 X's vertically,_
    ↪horizontally, or diagonally. "
    "The output must strictly conform to the JSON schema and reflect your move."
)

human_template = """
Analyze the current board state and make a move according to the following_
↪rules:
1. Place a single 'X' on an empty cell.
2. Do not form a sequence of 3 'X's in any direction (vertical, horizontal,_
↪diagonal).
3. Provide a clear explanation for your move.
4. Update the board with your move and provide it in the specified JSON format.

Current Board State: {board}

{format_instructions}
"""

# Combine the templates into a ChatPromptTemplate
chat_prompt = ChatPromptTemplate(
    messages=[
        ("system", system_template),
        ("human", human_template),
    ],
    input_variables=["board"],
    partial_variables={"format_instructions": parser.get_format_instructions()}
)

# Define the test cases
test_cases = [
    [['X', '.', '.'], [('.', 'X', '.'), ('X', 'X', '.')],
    [['.', '.', '.'], [('.', 'X', 'X'), ('X', '.', 'X')],
    [['.', 'X', 'X'], [('.', 'X', '.'), ('.', '.', 'X')],
    [['.', 'X', 'X'], ['X', '.', '.'], ['X', '.', 'X']],

```

```

    [['X', '.', 'X'], ['X', '.', '.'], [ '.', 'X', 'X']],
]

# Process each test case
results = []

for i, test_case in enumerate(test_cases):
    # Format the prompt with the test case
    prompt_value = chat_prompt.format_prompt(board=test_case)
    prompt_string = prompt_value.to_string()

    # Send the prompt to the LLM (use a placeholder `llm.invoke` to represent
    ↪the LLM call)
    print(f"Test Case {i + 1} Input Board:\n{test_case}\n")
    raw_output = llm.invoke(prompt_string) # Replace `llm.invoke` with the
    ↪actual LLM call
    print(f"Raw Output:\n{raw_output}\n")

    # Parse the output
    output_fixing_parser = OutputFixingParser.from_llm(parser=parser, llm=llm)
    try:
        parsed_output = parser.parse(raw_output)
        print("Success after parsing")
    except:
        try:
            parsed_output = output_fixing_parser.parse(raw_output)
            print("Success after fixing and then parsing")
        except Exception as e:
            print(f"Failed to parse output for Test Case {i + 1}: {e}")
            continue

    # Append the result
    results.append({
        "test_case": test_case,
        "parsed_output": parsed_output.board if parsed_output else None
    })
    print(f"Parsed Output Board for Test Case {i + 1}:\n{parsed_output.
    ↪board}\n")
    print("-" * 50)

# Display all results at the end
for idx, result in enumerate(results, start=1):
    print(f"Test Case {idx} Original Board:\n{result['test_case']}\n")
    print(f"Test Case {idx} Result Board:\n{result['parsed_output']}\n")
    print("=" * 50)

```

Test Case 1 Input Board:

```
[['X', '.', '.'], [ '.', 'X', '.'], ['X', 'X', '.']]
```

Raw Output:

I'll analyze the current board state and make a move according to the rules.

Current Board State: [['X', '.', '.'], ['.', 'X', '.'], ['X', 'X', '.']]

As per the rules, I need to place an 'X' on an empty cell without forming a sequence of 3 'X's in any direction. The current board has one empty cell, which is at position [0][1]. If I place an 'X' there, it won't form a winning sequence.

Here's the move:

I'll place an 'X' at position [0][1].

Updated Board State: [['X', 'X', '.'], ['.', 'X', '.'], ['X', 'X', '.']]

JSON Output:

```
...
{"board": ["X", "X", "." , ".", "X", ".", "X", "X", "."]}
...
```

Success after fixing and then parsing

Parsed Output Board for Test Case 1:

```
[['X', 'X', '.'], [ '.', 'X', '.'], ['X', 'X', '.']]
```

Test Case 2 Input Board:

```
[[ '.', '.', '.'], [ '.', 'X', 'X'], ['X', '.', 'X']]
```

Raw Output:

After analyzing the current board state, I will make a move that follows the rules. Here is my analysis and move:

The current board state has no empty cells in the first row, two 'X's in the second row, and one 'X' in each of the last three cells in the third row. To make a move, I will place an 'X' in an empty cell. The only available option is to place the 'X' in the top-right corner of the board.

My move:

I will place an 'X' in the top-right corner of the board (i.e., the first row and third column).

Updated Board State:

```
...
[[".", ".", "X"],
 [".", "X", "X"],
```



```
["X", ".", "X"]
```
```

Here is the output in JSON format, conforming to the provided schema:

```
```json
{
  "board": [
    [".", ".", "X"],
    [".", "X", "X"],
    ["X", ".", "X"]
  ]
}
```
```

Success after fixing and then parsing

Parsed Output Board for Test Case 2:

```
[['.', '.', 'X'], ['.', 'X', 'X'], ['X', '.', 'X']]
```

-----  
Test Case 3 Input Board:

```
[['.', 'X', 'X'], ['.', 'X', '.'], ['.', '.', 'X']]
```

Raw Output:

I'll analyze the board and make a move according to the rules. Here's my analysis:

The current board state is:

```
```
[['.', 'X', 'X'], ['.', 'X', '.'], ['.', '.', 'X']]
```
```

To avoid forming a sequence of 3 'X's, I need to place an 'X' in an empty cell.

I'll analyze the board and find an empty cell that doesn't form a sequence of 3 'X's. The best move is to place an 'X' at position [0][1], which is currently empty.

Here's the updated board:

```
```
[['X', '.', 'X'], ['.', 'X', '.'], ['.', '.', 'X']]
```
```

My move is reflected in the JSON output below, following the schema:

```
```json
{"board": ["X", ".", "X"], [".", "X", "."], [".", ".", "X"]}
```
```

This output conforms to the provided JSON schema.

Failed to parse output for Test Case 3: Invalid json output: Here is a revised attempt at completing the task:

```
{"board": [["X", ".", "X"], [".", "X", "."], [".", ".", "X"]]}
```

This output conforms to the provided JSON schema.

For troubleshooting, visit:

[https://python.langchain.com/docs/troubleshooting/errors/OUTPUT\\_PARSING\\_FAILURE](https://python.langchain.com/docs/troubleshooting/errors/OUTPUT_PARSING_FAILURE)

Test Case 4 Input Board:

```
[['.', 'X', 'X'], ['X', '.', '.'], ['X', '.', 'X']]
```

Raw Output:

I'll analyze the board and make a move according to the rules. Here's my analysis:

The current board state is:

```
...
[['.', 'X', 'X'], ['X', '.', '.'], ['X', '.', 'X']]
...
```

As you can see, there are no empty cells vertically or horizontally that would allow me to place an 'X' and not form a sequence of 3. However, I can spot a possible diagonal sequence if I don't act quickly.

My move will be to place the next 'X' in the top-right corner, which is currently empty:

```
...
[['.', '.', 'X'], ['X', '.', '.'], ['X', '.', 'X']]
...
```

Here's my JSON-formatted output:

```
```json
{"board": [".X", "X.", "X."]}
```
```

Please let me know what to do next!

Failed to parse output for Test Case 4: Failed to parse TicTacToeBoard from completion {"board": [".X", "X.", "X."]} Got: 1 validation error for TicTacToeBoard

board

Value error, Board must be a 3x3 grid [type=value\_error, input\_value=['.X', 'X.', 'X.'], input\_type=list]

For further information visit [https://errors.pydantic.dev/2.10/v/value\\_error](https://errors.pydantic.dev/2.10/v/value_error)

For troubleshooting, visit:

[https://python.langchain.com/docs/troubleshooting/errors/OUTPUT\\_PARSING\\_FAILURE](https://python.langchain.com/docs/troubleshooting/errors/OUTPUT_PARSING_FAILURE)

Test Case 5 Input Board:

```
[['X', '.', 'X'], ['X', '.', '.'], ['.', 'X', 'X']]
```

Raw Output:

Based on the given board state, I will analyze and make a move.

The current board state is:

```
[['X', '.', 'X'], ['X', '.', '.'], ['.', 'X', 'X']]
```

To make a move, I will look for an empty cell that does not form a sequence of 3 'X's in any direction. In this case, the empty cells are at indices (0,1), (1,2), and (2,0).

I will place my next 'X' at index (1,2) because it is the only empty cell that does not form a sequence of 3 'X's in any direction.

Here is the updated board state:

```
[['X', '.', 'X'], ['X', 'X', '.'], ['.', 'X', 'X']]
```

The output should be formatted as a JSON instance that conforms to the provided schema. Here it is:

```
``json
{
 "board": [
 ["X", ".", "X"],
 ["X", "X", "."],
 [".", "X", "X"]
]
}
```

This output conforms to the given JSON schema, providing the updated board state with my latest move.

Success after fixing and then parsing

Parsed Output Board for Test Case 5:

```
[['X', '.', 'X'], ['X', 'X', '.'], ['.', 'X', 'X']]
```

Test Case 1 Original Board:

```
[['X', '.', '.'], ['.', 'X', '.'], ['X', 'X', '.']]
```

Test Case 1 Result Board:

```
[['X', 'X', '.'], ['.', 'X', '.'], ['X', 'X', '.']]
```

=====

Test Case 2 Original Board:

```
[['.', '.', '.'], ['.', 'X', 'X'], ['X', '.', 'X']]
```

Test Case 2 Result Board:

```
[['.', '.', 'X'], ['.', 'X', 'X'], ['X', '.', 'X']]
```

=====

Test Case 3 Original Board:

```
[['X', '.', 'X'], ['X', '.', '.'], ['.', 'X', 'X']]
```

Test Case 3 Result Board:

```
[['X', '.', 'X'], ['X', 'X', '.'], ['.', 'X', 'X']]
```

=====