



Modules

Agents

Tools

Tools as OpenAI Functions

# Tools as OpenAI Functions

This notebook goes over how to use LangChain tools as OpenAI functions.

```
%pip install -qU langchain-community  
langchain-openai
```

```
from langchain_community.tools import  
MoveFileTool  
from langchain_core.messages import  
HumanMessage  
from langchain_core.utils.function_calling  
import convert_to_openai_function  
from langchain_openai import ChatOpenAI
```

```
model = ChatOpenAI(model="gpt-3.5-turbo")
```

```
tools = [MoveFileTool()]  
functions = [convert_to_openai_function(t)
```

```
for t in tools]
```

```
functions[0]
```

```
{'name': 'move_file',  
  'description': 'Move or rename a file from  
one location to another',  
  'parameters': {'type': 'object',  
    'properties': {'source_path':  
{'description': 'Path of the file to move',  
    'type': 'string'},  
    'destination_path': {'description': 'New  
path for the moved file',  
    'type': 'string'}}},  
  'required': ['source_path',  
    'destination_path']}]}
```

```
message = model.invoke(  
    [HumanMessage(content="move file foo to  
bar")], functions=functions  
)
```

```
message
```

```
AIMessage(content='', additional_kwargs={
    'function_call': {'arguments': '{\n
    "source_path": "foo",\n  "destination_path":
    "bar"\n}', 'name': 'move_file'}})
```

```
message.additional_kwargs["function_call"]
```

```
{'name': 'move_file',
  'arguments': '{\n  "source_path": "foo",\n
  "destination_path": "bar"\n}'}
```

With OpenAI chat models we can also automatically bind and convert function-like objects with `bind_functions`

```
model_with_functions = model.bind_functions(too
model_with_functions.invoke([HumanMessage(conte
file foo to bar"))])
```

```
AIMessage(content='', additional_kwargs={
    'function_call': {'arguments': '{\n
    "source_path": "foo",\n  "destination_path":
    "bar"\n}', 'name': 'move_file'}})
```

Or we can use the update OpenAI API that uses `tools` and `tool_choice` instead of `functions` and `function_call` by using `ChatOpenAI.bind_tools`:

```
model_with_tools = model.bind_tools(tools)
model_with_tools.invoke([HumanMessage(content="I
file foo to bar")])
```

```
AIMessage(content='', additional_kwargs=
{'tool_calls': [{'id':
'call_btkY3xV71cEVA0HnNa5qwo44', 'function':
{'arguments': '{\n  "source_path": "foo",\n
"destination_path": "bar"\n}', 'name':
'move_file'}, 'type': 'function'}]})
```