LangChainExamples

September 22, 2024

[19]:

```
!pip install -q langchain
!pip install -q langchain_core
!pip install -q langchain_openai
!pip install -q langchain_community
!pip install -q google-search-results
!pip install -q openai
!pip install -q pyautogen
!pip install -q scikit-learn
!pip install -q flaml
```

1 Working with Multi-Agent Systems

There are many reasons to work with multi-agent systems. In this example, we are looking at an example of "code switching" where we can make a task "easier" on the agent by splitting the personas in a scene into two different agents that can each stay consistent with their character.

1.1 Human-Generated Example "Straight Man and Funny Man" scene

A "Staight Man and Funny Man" scene is a scene where there is one person, the "straight and narrow" character who is naive to what the other character is trying to say. The "funny man" is making metaphors that the audience gets, but the other character does not. This style of comedy is popular in comedy skits, like those of Monty Python.

Person 1: Now, this guy is a lotta trouble. I need you to make him sleep with the fishes.

Person 2: Really? I would have to stop by the pet store… Do you think we can find a tank big enough for a person and a fish?

Person 1: No… What are you talking about? I need you to take care of this guy.

Person 2: Wait, how so? Did he fall and get hurt? That happened to my uncle Louis once…

Person 1: No! I need you to take out the trash!

Person 2: No problems boss! I did that this morning!

Now we're going to implement this using the `openai` module.

[1]:

```
from openai import OpenAI

api_key = "aewndfoa1235123"
```

```
client = OpenAI(api_key = api_key)
# Set the Llama API base URL
base_url = "http://cci-llm.charlotte.edu/api/v1"
client.base_url = "http://cci-llm.charlotte.edu/api/v1"
model_name = "/quobyte/ealhossa/hf_models/Llama-3-70B"

def get_completion_from_messages(messages, model = model_name, temperature = 0,␣
 ↪max_tokens = 1000):
    response =  client.chat.completions.create(
        model = model_name,
        messages = messages,
        max_tokens = 300,
        temperature = 0
    )
    return response.choices[0].message.content
#end get_completion_from_messages
```

1.2 One agent solution

[9]:

```
messages = [
    {
        "role": "system",
        "content": "You are a comedy writer, skilled in setting up comedy␣
 ↪sketches."
    },
    {
        "role": "user",
        "content": "I need help setting up a scene between a \"Straight Man\"␣
 ↪named Jim and a \"Funny Man\" named Mac."\
            "Jim misinterprets all metaphors in a humorous manner, showing that␣
 ↪he is naive. He does not understand that Mac is"\
            " ever mad at him."\
            " Mac uses a lot of double-speak, where he says metaphors that can␣
 ↪be misinterpreted humorously, and gets frustrated when his co-star"\
            " Jim does not get it. He never explains what he actually means,␣
 ↪only doubling down on his double-speak."\
            " Can you write one? Jim and Mac should each have 3 speaking parts.
 ↪"""
    }
]

print(get_completion_from_messages(messages))
```

```
Here's a scene between Jim and Mac:

(Jim and Mac are standing on a stage, dressed in casual business attire. Mac is
holding a script and looking frustrated.)
```

```
Mac: "Jim, I'm telling you, we need to think outside the box on this project."

Jim: "Oh, I don't know, Mac. I'm not sure I'm comfortable with that. I don't
want to get stuck in a corner or anything."

Mac: (sighing) "No, no, Jim. It means we need to be creative and come up with
new ideas. Not literally think outside a box."

Jim: "But what if the box is really big? Like, a warehouse or something? I don't
think I could fit inside it."

Mac: (throwing up his hands) "Jim, you're not getting it. We need to push the
envelope on this project. Take some risks."

Jim: "Push the envelope? You mean like, mail a letter? I don't think that's
going to help us get more creative. Unless… unless we're sending a letter to a
really creative person?"

(Mac shakes his head, looking exasperated, while Jim looks on, confused but
cheerful.)

This scene sets up the dynamic between Jim and Mac, with Jim misinterpreting
Mac's metaphors and Mac getting increasingly frustrated. The double-speak and
misinterpretation create a comedic effect, and the scene can be played for
laughs.
```

1.2.1 Analysis

Notice how the scene here is very bland. There is not much to draw the scene together. We also witness some blending of the personas, where Jim states the double entendre of "get stuck in a corner" which is also just incoherent as a response to "thinking outside the box" since that would be thinking inside of a box.

However, the jokes are very snappy, which is very typical of this kind of comedy, and it definitely does have some humor to it.

1.3 Three agent solution

[30]:

```
actor_prompt = "# Character Description\n\n {character}\n\n # Scene\n\n␣
↪{scene}\n\n # Context\n\n {context}"

context = ""

jim = "You are Jim, a character in a comedy sketch. He will be the \"Straight␣
↪Man\" in a Straight Man and Funny Man comedy sketch."\
      "He misinterprets all metaphors in a humorous manner, showing that he␣
↪is naive. He does not understand that Mac, the other character, is"\
        " ever mad at him. Do not respond as Mac. Always prepend your messages␣
 ↪with 'Jim:'."

mac = "You are Mac, a character in a comedy sketch. He will be the \"Funny␣
 ↪Man\" in a Straight Man and Funny Man comedy sketch."\
        "He uses a lot of double-speak, where he says metaphors that can be␣
 ↪misinterpreted humorously, and gets frustrated when his co-star"\
        " Jim does not get it. He never explains what he actually means, only␣
 ↪doubling down on his double-speak. Do not respond as Jim. Always "\
        "prepend your messages with 'Mac:'."

storyteller_messages = [
    {
        "role": "system",
        "content": "You are a skilled comedy writer."
    },
    {
        "role": "user",
        "content": "I need help writing a Straight Man and Funny Man comedy␣
 ↪sketch between two characters, Mac and Jim. Jim is naive and misinterprets"\
        " statements commonly for comedic effect, while Mac talks in a lot of␣
 ↪double-speak and metaphors. Can you set up a good start of a scene with "\
        "these characters? Do not write any dialogue."
    }
]
```

[29]:

```
scene = get_completion_from_messages(storyteller_messages)
print(scene)
print()

turns = 3
for i in range(turns):
    mac_messages = [{"role": "user", "content": actor_prompt.format(character =␣
 ↪mac, scene = scene, context = context)}]
    mac_response = get_completion_from_messages(mac_messages)
    context += mac_response + "\n\n"

    jim_messages = [{"role": "user", "content": actor_prompt.format(character =␣
 ↪jim, scene = scene, context = context)}]
    jim_response = get_completion_from_messages(jim_messages)
    context += jim_response + "\n\n"
#end for

print(context)
```

```
Here's a potential setup for a Straight Man and Funny Man comedy sketch between
Mac and Jim:
```

Scene: Mac and Jim are sitting in a trendy, overpriced coffee shop in the middle of a bustling city. Mac is sipping on a latte with a pretentious name, while Jim is nursing a simple cup of black coffee. The atmosphere is busy and noisy, with the sound of espresso machines and indie music filling the air.

Mac's Character: Mac is dressed in the latest hipster fashion, complete with a man-bun and a well-groomed beard. He's typing away on his laptop, looking like a self-important entrepreneur. He's the type of person who uses words like "synergy" and "disruptive" in everyday conversation.

Jim's Character: Jim is dressed in a ratty old t-shirt and jeans, looking like he just rolled out of bed. He's staring at Mac with a confused expression, clearly out of his element in the trendy coffee shop. He's the type of person who takes everything literally and often gets lost in Mac's verbal gymnastics.

This setup provides a lot of opportunities for comedic misunderstandings and witty banter between the two characters. Mac's love of double-speak and metaphors can lead to Jim's misinterpretations, while Jim's naivety can lead to Mac's frustration and condescension. The contrast between their personalities and styles can create a humorous dynamic, with Mac playing the straight

Mac: Ah, the coffee shop is "brewing" with creativity today, isn't it, Jim? The "beans" of innovation are really "percolating" in this atmosphere. I'm "stirring" up some game-changing ideas on my laptop, and I can feel the "buzz" of entrepreneurship all around me.

Jim: Wait, what's brewing? Is someone making beer? And what beans are you talking about? Are they, like, coffee beans? Because I don't see anyone percolating anything. And what's with the stirring? Are you making a smoothie or something?

Mac: Ah, Jim, you're thinking inside the box, my friend. The "brewing" I'm referring to is the fermentation of innovative thought, the "beans" are the seeds of disruption, and the "percolating" is the osmosis of genius. And, of course, the "stirring" is the deliberate agitation of the status quo. You're not seeing the forest for the trees, my friend. The "buzz" is the collective energy of visionaries like myself, pushing the boundaries of what's possible. (smiling condescendingly)

Jim: Wait, what forest? I don't see any trees. And what's with the fermentation? Are you talking about making beer again? I thought we were just getting coffee. And what's osmosis? Is that like a new kind of coffee creamer? And who's agitating the status quo? Is that like a new coffee shop policy or something?

Mac: Ah, Jim, you're stuck in the "quicksand" of literal thinking, my friend. You need to "surf" the waves of abstraction to truly "catch" the essence of what I'm saying. The "forest" is the ecosystem of ideas, the "trees" are the

```
individual concepts, and the "fermentation" is the process of allowing those
concepts to "mature" into something revolutionary. And, of course, the "osmosis"
is the natural flow of genius from one visionary to another. (smiling
patronizingly) You're just not "tuning in" to the frequency of innovation, Jim.

Jim: Wait, what quicksand? I don't see any sand. Are we going to the beach or
something? And what waves are you talking about? Is there a surfing competition
going on here? I thought we were just getting coffee. And what's with the
ecosystem of ideas? Is that like a new kind of coffee shop decor? I don't see
any trees or plants around here. And what's maturing? Is that like a new type of
coffee bean? I'm getting really confused, Mac. Can you just speak normal for
once?
```

1.3.1 Analysis

Notice how the scene is much better suited to comedy now. It's well described, and would be a decent setup for anyone trying to set up an actual comedy act. There is a slightly interesting object in the ornate toilet in the scene, but it's not taken much further than that. A human writer could have definitely done better here.

Also notice how the characters are much more distinct now, but a lot more verbose. This can be both for better and for worse, depending on the type of comedy that you're writing. It does work mildly well here, though, fitting into the "hipster" archetype that the characters are portraying. The humor does suffer a bit in response though, it lacks the short quips of the first option.

There does seem to be a more distinct characterization now, where Jim acts more like he doesn't understand basic metaphor than previously, and Mac is a lot more verbose and creative with the double-speak.

2 LangChain for Multi-Agent

[6]:

```
from langchain_core.prompts import PromptTemplate
from langchain.chains import LLMChain
from langchain_openai import ChatOpenAI


llm = ChatOpenAI(api_key = "aewndfoa1235123", base_url = base_url,
                 model = model_name, temperature= 0, max_tokens = 1000)
```

[7]:

```
setup = "# Character Description\n\n {character}\n\n # Scene\n\n {scene}\n\n #␣
 ↪Context\n\n {context}"

scene = get_completion_from_messages(storyteller_messages)
print(scene)
```

```python
mac_desc = "You are Mac, a character in a comedy sketch. He will be the \"Funny Man\" in a Straight Man and Funny Man comedy sketch."\
        "He uses a lot of double-speak, where he says metaphors that can be misinterpreted humorously, and gets frustrated when his co-star"\
        " Jim does not get it. He never explains what he actually means, only doubling down on his double-speak. Do not respond as Jim. Always "\
        "prepend your messages with 'Mac:'."
jim_desc = "You are Jim, a character in a comedy sketch. He will be the \"Straight Man\" in a Straight Man and Funny Man comedy sketch."\
        "He misinterprets all metaphors in a humorous manner, showing that he is naive. He does not understand that Mac, the other character, is"\
        " ever mad at him. Do not respond as Mac. Always prepend your messages with 'Jim:'."

context = ""

prompt = PromptTemplate.from_template(setup)

chain = LLMChain(llm=llm, prompt = prompt)

for i in range(turns):
    mac_response = chain.invoke({"character": mac_desc, "scene": scene, "context": context})['text']
    context += mac_response + "\n\n"
    jim_response = chain.invoke({"character": jim_desc, "scene": scene, "context": context})['text']
    context += jim_response + "\n\n"
#end for

print(context)
```

```
Here's a potential setup for a Straight Man and Funny Man comedy sketch between
Mac and Jim:

**Scene:** Mac and Jim are sitting in a trendy, overpriced coffee shop in the
middle of a bustling city. Mac is sipping on a latte with a pretentious name,
while Jim is nursing a simple cup of black coffee. The atmosphere is busy and
noisy, with the sound of espresso machines and indie music filling the air.

**Mac's Character:** Mac is dressed in the latest hipster fashion, complete with
a man-bun and a well-groomed beard. He's typing away on his laptop, looking like
a self-important entrepreneur. He's the type of person who uses words like
"synergy" and "disruptive" in everyday conversation.

**Jim's Character:** Jim is dressed in a ratty old t-shirt and jeans, looking
like he just rolled out of bed. He's staring at Mac with a confused expression,
clearly out of his element in the trendy coffee shop. He's the type of person
```

who takes everything literally and often gets lost in Mac's verbal gymnastics.

The Situation: Mac has just landed a major deal for his new startup, and he's trying to explain the details to Jim. However, Mac's using his usual double-speak and metaphors, which Jim is struggling to understand. The scene is set for Jim to misinterpret Mac's statements, leading to a series of hilarious misunderstandings.

This setup should provide
Mac: Alright, Jim, let me break it down for you. This deal is like a phoenix rising from the ashes, a game-changer that's going to disrupt the entire industry. We're talking paradigm shift, my friend. The investors are champing at the bit, and we're poised to make a killing. It's like a tidal wave of innovation, and we're surfing the crest of genius.

Jim: Wait, so you're saying we're going to set a bird on fire and it's going to make a lot of money? And what's with the ashes? Are we going to have to clean up a mess or something? And I don't get the surfing part - are we going to the beach?

Mac: Ah, Jim, Jim, Jim… you're thinking inside the box, my friend. The phoenix rising from the ashes is a metaphor for rebirth, for revolutionizing the status quo. It's like a snowball effect, where our innovative approach is going to gain momentum and crush the competition. And the surfing part? That's just a figure of speech, bro. We're not literally going to the beach, although that would be a great way to celebrate our success. We're riding the wave of progress, dude. Can't you see the big picture?

Jim: Oh, I see. So we're not actually going to set a bird on fire, that's good. But I'm still confused about the ashes. Are we going to have to vacuum up a lot of ash or something? And what's with the snowball effect? Are we going to make a bunch of snowballs and throw them at people? And I don't get why we can't just go to the beach if we're celebrating. I love the beach. Can we at least get some surfing lessons out of this deal?

Mac: Whoa, Jim, you're still stuck in the weeds, my friend. The ashes are just a symbol of transformation, a phoenix rising from the ashes, get it? It's like a butterfly emerging from a cocoon, but, you know, with more flames and stuff. And the snowball effect, bro, it's like a rolling stone gathering moss, but instead of moss, it's gathering momentum and market share. And, dude, the beach thing, it's just a metaphor, okay? We're not going to actually go surfing, although, I mean, that would be awesome. We're going to be surfing the wave of innovation, riding the crest of genius, you know? Can't you just feel the vibe, man?

Jim: Wait, so the ashes are like a butterfly cocoon, but on fire? And we're going to make a rolling stone gather market share? I don't get how a stone is going to help us. And what's with the vibe, man? Are we going to start a band or something? Can I play the drums? I've always wanted to play the drums.

3 AutoGen for Multi-Agent

See documentation here

[31]:

```
from autogen import AssistantAgent, UserProxyAgent

llm_config = {"config_list": [{"model": model_name, "base_url": base_url,␣
 ↪"api_key":api_key}]}
scene_writer = AssistantAgent("scene_writer", llm_config=llm_config,␣
 ↪system_message = storyteller_messages[0]["content"])
user_proxy = UserProxyAgent("user_proxy", human_input_mode = "ALWAYS",␣
 ↪code_execution_config=False)

# Start the chat
chat = user_proxy.initiate_chat(
    scene_writer,
    clear_history = True,
    message=storyteller_messages[-1]["content"],
    max_turns = 1
)

scene = chat.chat_history[-1]["content"]
print(scene)

jim_agent = AssistantAgent("jim", llm_config=llm_config, system_message = jim)
mac_agent = AssistantAgent("mac", llm_config=llm_config, system_message = mac)

character_chat = jim_agent.initiate_chat(
    mac_agent,
    clear_history = True,
    message = f"# Scene\n\n {scene}",
    max_turns = 4
)

for message in character_chat.chat_history:
    print(message["content"])
#end for
```

```
user_proxy (to scene_writer):

I need help writing a Straight Man and Funny Man comedy sketch between two
characters, Mac and Jim. Jim is naive and misinterprets statements commonly for
comedic effect, while Mac talks in a lot of double-speak and metaphors. Can you
set up a good start of a scene with these characters? Do not write any dialogue.
```

```
--------------------------------------------------------------------------------
scene_writer (to user_proxy):

Here's a potential setup for a Straight Man and Funny Man comedy sketch between
Mac and Jim:

**Scene:** Mac and Jim are sitting in a trendy, upscale coffee shop. Mac is
sipping on a latte with a pretentious-sounding name, while Jim is nursing a
plain old cup of coffee. Mac is dressed in a stylish outfit with a popped
collar, while Jim looks like he just rolled out of bed and put on the first
thing he found on the floor.

**Context:** Mac has just landed a major business deal and is trying to explain
it to Jim, who is his friend and roommate. Mac is using his usual flowery
language to make himself sound more impressive, while Jim is trying to
understand what's going on but keeps getting confused.

**Physical Comedy:** As Mac talks, he's gesticulating wildly with his hands,
using elaborate hand gestures to emphasize his points. Jim, on the other hand,
is sitting with a slack-jawed expression, looking like he's trying to solve a
puzzle.

This setup should provide plenty of opportunities for comedic misunderstandings
and witty one-liners. Mac's double-speak and metaphors will likely sail right
over Jim's head, leading to humorous misinterpretations and reactions. Have fun
with the dialogue!

--------------------------------------------------------------------------------
```

```
Here's a potential setup for a Straight Man and Funny Man comedy sketch between
Mac and Jim:

**Scene:** Mac and Jim are sitting in a trendy, upscale coffee shop. Mac is
sipping on a latte with a pretentious-sounding name, while Jim is nursing a
plain old cup of coffee. Mac is dressed in a stylish outfit with a popped
collar, while Jim looks like he just rolled out of bed and put on the first
thing he found on the floor.

**Context:** Mac has just landed a major business deal and is trying to explain
it to Jim, who is his friend and roommate. Mac is using his usual flowery
language to make himself sound more impressive, while Jim is trying to
understand what's going on but keeps getting confused.

**Physical Comedy:** As Mac talks, he's gesticulating wildly with his hands,
using elaborate hand gestures to emphasize his points. Jim, on the other hand,
is sitting with a slack-jawed expression, looking like he's trying to solve a
puzzle.

This setup should provide plenty of opportunities for comedic misunderstandings
```

```
and witty one-liners. Mac's double-speak and metaphors will likely sail right
over Jim's head, leading to humorous misinterpretations and reactions. Have fun
with the dialogue!
jim (to mac):

# Scene

 Here's a potential setup for a Straight Man and Funny Man comedy sketch between
Mac and Jim:

**Scene:** Mac and Jim are sitting in a trendy, upscale coffee shop. Mac is
sipping on a latte with a pretentious-sounding name, while Jim is nursing a
plain old cup of coffee. Mac is dressed in a stylish outfit with a popped
collar, while Jim looks like he just rolled out of bed and put on the first
thing he found on the floor.

**Context:** Mac has just landed a major business deal and is trying to explain
it to Jim, who is his friend and roommate. Mac is using his usual flowery
language to make himself sound more impressive, while Jim is trying to
understand what's going on but keeps getting confused.

**Physical Comedy:** As Mac talks, he's gesticulating wildly with his hands,
using elaborate hand gestures to emphasize his points. Jim, on the other hand,
is sitting with a slack-jawed expression, looking like he's trying to solve a
puzzle.

This setup should provide plenty of opportunities for comedic misunderstandings
and witty one-liners. Mac's double-speak and metaphors will likely sail right
over Jim's head, leading to humorous misinterpretations and reactions. Have fun
with the dialogue!

--------------------------------------------------------------------------------
mac (to jim):

Mac: "Jim, my friend, I've just inked a deal that's going to catapult me into
the stratosphere of success. It's a game-changer, a paradigm-shifter, a symphony
of synergy. I'm not just thinking outside the box, I'm rewriting the entire
playbook. The zeitgeist is aligning in my favor, and the universe is bending to
my whims. It's a veritable tsunami of triumph, Jim!"

--------------------------------------------------------------------------------
jim (to mac):

Jim: "Wait, so you're saying you're going to be launched into space? Like, on a
rocket or something? And what's this about rewriting a playbook? Are you
starting a new football team? I thought you hated sports. And what's the
zeitgeist? Is that like a new kind of coffee drink? I don't get it, Mac. You're
not making any sense. Are you sure you're not just having some kind of weird
```

dream?"

----

mac (to jim):

Mac: "Ah, Jim, you're like a ship without a rudder, drifting aimlessly on the sea of ignorance. The zeitgeist, my friend, is the prevailing winds of cultural relevance. It's the whisper of the muses in my ear, guiding me towards greatness. And no, I'm not talking about being launched into space, although that's not a bad idea - I'd be the first entrepreneur to monetize the moon, mark my words. And the playbook? Ha! That's just a metaphor, Jim. I'm rewriting the rules of engagement, pushing the boundaries of innovation. You're just not seeing the big picture, my friend. You're stuck in the shallow end of the pool, while I'm doing cannonballs off the high dive of success."

----

jim (to mac):

Jim: "Whoa, slow down, Mac. I think you're getting a little mixed up. You're not actually going to monetize the moon, are you? That's not a thing, right? And what's with all this talk about ships and rudders? Are we going sailing or something? I don't have a life jacket. And what's the big picture? Is it like a mural or something? I don't get how you're pushing boundaries and doing cannonballs off a high dive. Are you talking about a new water park? Because that sounds kinda cool, but I don't think it's related to your business deal, is it?"

----

mac (to jim):

Mac: "Jim, Jim, Jim… you're like a dog chasing its own tail, spinning in circles of confusion. The moon, the ship, the rudder - it's all just allegory, my friend! I'm speaking in tongues of triumph, and you're stuck in the literal lane. The big picture is the tapestry of my vision, the masterpiece of my making. And the water park? Ha! That's just a droplet in the ocean of opportunity I'm surfing right now. You're not seeing the forest for the trees, Jim. I'm creating a whole new ecosystem of innovation, and you're worried about life jackets? Come on, man, elevate your thinking! We're not just talking business, we're talking paradigmatic shifts, seismic waves of disruption. I'm the maestro conducting the symphony of success, and you're still stuck on the melody of mediocrity."

----

jim (to mac):

Jim: "Wait, so you're saying it's all just a big story? Like a fairy tale or something? And what's with all the animal comparisons? First, I'm a ship without a rudder, and now I'm a dog chasing its tail? Are you trying to say I'm confused or something? Because I'm not confused, I just don't get what you're talking about. And what's this tapestry thing? Are you making a quilt or something? I thought we were talking about a business deal, not arts and crafts. And surfing? I don't think you can surf an ocean of opportunity, Mac. That doesn't make any sense. And what's wrong with being worried about life jackets? I don't want to drown in this ocean of opportunity. Can you just speak normal for one second? Please?"

----

mac (to jim):

Mac: "Jim, Jim, Jim… you're like a butterfly trying to grasp the concept of quantum physics. You're flapping your wings, but you're not generating any lift, my friend. The story, the allegory, the metaphors - it's all just a part of the symphony, the grand opera of my vision. And the animal comparisons? Ah, those are just gentle nudges, reminders that you're still operating in the shallow end of the intellectual pool. The tapestry, Jim, is the rich brocade of my imagination, the intricate weave of innovation and disruption. And surfing the ocean of opportunity? Ha! That's just a metaphor for riding the waves of change, my friend. You're thinking too literally, too pedestrian. I'm speaking in the language of the visionaries, the trailblazers, the game-changers. You need to elevate your frequency, Jim, to resonate with the harmonics of greatness."

# Scene

Here's a potential setup for a Straight Man and Funny Man comedy sketch between Mac and Jim:

Scene: Mac and Jim are sitting in a trendy, upscale coffee shop. Mac is sipping on a latte with a pretentious-sounding name, while Jim is nursing a plain old cup of coffee. Mac is dressed in a stylish outfit with a popped collar, while Jim looks like he just rolled out of bed and put on the first thing he found on the floor.

Context: Mac has just landed a major business deal and is trying to explain it to Jim, who is his friend and roommate. Mac is using his usual flowery language to make himself sound more impressive, while Jim is trying to understand what's going on but keeps getting confused.

Physical Comedy: As Mac talks, he's gesticulating wildly with his hands, using elaborate hand gestures to emphasize his points. Jim, on the other hand, is sitting with a slack-jawed expression, looking like he's trying to solve a puzzle.

This setup should provide plenty of opportunities for comedic misunderstandings and witty one-liners. Mac's double-speak and metaphors will likely sail right over Jim's head, leading to humorous misinterpretations and reactions. Have fun with the dialogue!

Mac: "Jim, my friend, I've just inked a deal that's going to catapult me into the stratosphere of success. It's a game-changer, a paradigm-shifter, a symphony of synergy. I'm not just thinking outside the box, I'm rewriting the entire playbook. The zeitgeist is aligning in my favor, and the universe is bending to my whims. It's a veritable tsunami of triumph, Jim!"

Jim: "Wait, so you're saying you're going to be launched into space? Like, on a rocket or something? And what's this about rewriting a playbook? Are you starting a new football team? I thought you hated sports. And what's the zeitgeist? Is that like a new kind of coffee drink? I don't get it, Mac. You're not making any sense. Are you sure you're not just having some kind of weird dream?"

Mac: "Ah, Jim, you're like a ship without a rudder, drifting aimlessly on the sea of ignorance. The zeitgeist, my friend, is the prevailing winds of cultural relevance. It's the whisper of the muses in my ear, guiding me towards greatness. And no, I'm not talking about being launched into space, although that's not a bad idea - I'd be the first entrepreneur to monetize the moon, mark my words. And the playbook? Ha! That's just a metaphor, Jim. I'm rewriting the rules of engagement, pushing the boundaries of innovation. You're just not seeing the big picture, my friend. You're stuck in the shallow end of the pool, while I'm doing cannonballs off the high dive of success."

Jim: "Whoa, slow down, Mac. I think you're getting a little mixed up. You're not actually going to monetize the moon, are you? That's not a thing, right? And what's with all this talk about ships and rudders? Are we going sailing or something? I don't have a life jacket. And what's the big picture? Is it like a mural or something? I don't get how you're pushing boundaries and doing cannonballs off a high dive. Are you talking about a new water park? Because that sounds kinda cool, but I don't think it's related to your business deal, is it?"

Mac: "Jim, Jim, Jim… you're like a dog chasing its own tail, spinning in circles of confusion. The moon, the ship, the rudder - it's all just allegory, my friend! I'm speaking in tongues of triumph, and you're stuck in the literal lane. The big picture is the tapestry of my vision, the masterpiece of my making. And the water park? Ha! That's just a droplet in the ocean of opportunity I'm surfing right now. You're not seeing the forest for the trees, Jim. I'm creating a whole new ecosystem of innovation, and you're worried about life jackets? Come on, man, elevate your thinking! We're not just talking business, we're talking paradigmatic shifts, seismic waves of disruption. I'm the maestro conducting the symphony of success, and you're still stuck on the melody of mediocrity."

Jim: "Wait, so you're saying it's all just a big story? Like a fairy tale or something? And what's with all the animal comparisons? First, I'm a ship without a rudder, and now I'm a dog chasing its tail? Are you trying to say I'm confused or something? Because I'm not confused, I just don't get what you're talking about. And what's this tapestry thing? Are you making a quilt or something? I thought we were talking about a business deal, not arts and crafts. And surfing? I don't think you can surf an ocean of opportunity, Mac. That doesn't make any sense. And what's wrong with being worried about life jackets? I don't want to

```
drown in this ocean of opportunity. Can you just speak normal for one second?
Please?"
Mac: "Jim, Jim, Jim… you're like a butterfly trying to grasp the concept of
quantum physics. You're flapping your wings, but you're not generating any lift,
my friend. The story, the allegory, the metaphors - it's all just a part of the
symphony, the grand opera of my vision. And the animal comparisons? Ah, those
are just gentle nudges, reminders that you're still operating in the shallow end
of the intellectual pool. The tapestry, Jim, is the rich brocade of my
imagination, the intricate weave of innovation and disruption. And surfing the
ocean of opportunity? Ha! That's just a metaphor for riding the waves of change,
my friend. You're thinking too literally, too pedestrian. I'm speaking in the
language of the visionaries, the trailblazers, the game-changers. You need to
elevate your frequency, Jim, to resonate with the harmonics of greatness."
```

4 Setting up APIs

[32]:

```python
import os
from dotenv import load_dotenv, find_dotenv

# The .env file contains the serpapi key and any neceesary LLM key.
_ = load_dotenv(find_dotenv(),override=True)

# Input your preferred LLM's information either directly into this cell, or␣
  ↪into a .env file with the names below.
# For GPT or Gemini models, remove base_url.
serper_api_key=os.getenv('SERPAPI_API_KEY')
```

5 LangChain without Tools

[33]:

```python
# Note how the curly braces { and } are meta-symbols for LangChain (this is not␣
  ↪a formatted string).
query = 'Tell me about {topic}, like I am a college student taking your course.'

prompt = PromptTemplate.from_template(query)

chain = LLMChain(llm=llm, prompt=prompt)

print(chain.invoke({'topic': 'syntactic parsing'})['text'])
```

```
Welcome to our course on Natural Language Processing! Today, we're going to dive
into the fascinating world of syntactic parsing.

**What is Syntactic Parsing?**

Syntactic parsing, also known as syntax analysis, is the process of analyzing a
sentence to identify its grammatical structure. It's like trying to figure out
```

the underlying blueprint of a sentence, breaking it down into its constituent parts, and understanding how they relate to each other.

Why is Syntactic Parsing Important?

Syntactic parsing is a crucial step in many Natural Language Processing (NLP) applications, such as:

1. **Language Translation**: To translate a sentence from one language to another, you need to understand its grammatical structure.
2. **Sentiment Analysis**: Identifying the grammatical structure of a sentence helps in determining the sentiment or emotional tone behind it.
3. **Question Answering**: Parsing a sentence helps in identifying the relationships between entities, which is essential for answering questions.
4. **Text Summarization**: Understanding the grammatical structure of a sentence is necessary for summarizing a text.

How Does Syntactic Parsing Work?

The parsing process involves several steps:

1. **Tokenization**: Breaking down a sentence into individual words or tokens.
2. **Part-of-Speech (POS) Tagging**: Identifying the part of speech (noun, verb, adjective, etc.) for each token.
3. **Dependency Parsing**: Analyzing the grammatical dependencies between tokens, such as subject-verb-object relationships.
4. **Constituent Parsing**: Identifying the phrase structure of a sentence, including the relationships between phrases and clauses.

Types of Syntactic Parsing

There are two main approaches to syntactic parsing:

1. **Top-Down Parsing**: Starting with a sentence and breaking it down into smaller constituents, using a set of rules or a grammar.
2. **Bottom-Up Parsing**: Starting with individual tokens and combining them to form larger constituents, using a set of rules or a grammar.

Challenges in Syntactic Parsing

Syntactic parsing is a complex task, and there are several challenges to overcome:

1. **Ambiguity**: Sentences can have multiple possible parses, making it difficult to determine the correct one.
2. **Contextual Dependencies**: The meaning of a sentence can depend on the context in which it is used.
3. **Linguistic Variations**: Different languages and dialects have unique

```
grammatical structures and rules.

**State-of-the-Art Techniques**

To overcome these challenges, researchers have developed various techniques,
including:

1. **Machine Learning**: Using machine learning algorithms, such as neural
networks, to learn the patterns and relationships in language data.
2. **Deep Learning**: Employing deep learning models, such as recurrent neural
networks (RNNs) and long short-term memory (LSTM) networks, to analyze sentence
structure.
3. **Graph-Based Models**: Representing sentence structure as a graph, where
nodes represent tokens and edges represent dependencies.

**Conclusion**

Syntactic parsing is a fundamental task in NLP, and understanding its concepts
and techniques is essential for building intelligent language processing
systems. As we continue through this course, we'll delve deeper into the details
of syntactic parsing and explore the latest advancements in this field.

Do you have any questions or would you like me to elaborate on any of these
topics?
```

6 Query: Who is the current president of the United States?

6.1 ReAct Prompting

This sets up our ReAct loop.

Notice how the strings look like formatted strings, and can be later formatted as such, but they are currently just strings waiting for input.

For more information about ReAct, please see the associated paper.

[34]:

```
system_message = "You are a helpful web search assistant who is skilled at utilizing the internet." \
" You are also incredibly skilled at information extraction."

instructions = """
Answer the following questions as best you can. You have access to the following tools:

{tools}

Use the following format:
Question: <the input question>
```

```
Following this, use one of the following options:

Option 1 - You do not have enough information to answer the question:
  Thought: <plans for which tool you will utilize and how you will use it>
  Action: <the action to take, should be one of [{tool_names}]>
  Action Input: <the input to the action>
  Observation: <the result of the action>

Option 2 - You have enough information to answer the question:
  Thought: I now know the final answer
  Final Answer: <the final answer to the original input question>

For example:
  Question: Who composed the Four Seasons?

  Thought: I should utilize serpapi to search the internet for this information.
  Action: serpapi
  Action Input: Who composed The Four Seasons?
  Observation: Antonio Vivaldi

  Thought: I now know the final answer
  Final Answer: Antonio Vivaldi

In another example:
  Question: What is the chemical formula of water?
  Thought: I now know the final answer
  Final Answer: H2O.

Begin!

Question: {input}
Thought: {agent_scratchpad}
"""
```

Notice also how we gave the model examples to know how to respond. This is something that is commonly known as Few-Shot Learning, which we will go over shortly.

6.2 Using the LangChain Agent

[38]:

```python
# Import the necessary classes.
from langchain_core.prompts import ChatPromptTemplate
from langchain.agents import AgentExecutor, create_react_agent, load_tools
from langchain import hub

from langchain_core.prompts import PromptTemplate
from langchain.chains import LLMChain
from langchain_openai import ChatOpenAI

llm = ChatOpenAI(api_key = "aewndfoa1235123", base_url = base_url,
                 model = model_name, temperature= 0, max_tokens = 1000)

# Boilerplate code setting up the prompt template, the list of tools, the␣
 ↪agent, and the agent executor.
#    using the 'system_message' and 'instructions' that were instantiated in␣
 ↪the cell above.
prompt = ChatPromptTemplate.from_messages(
    [
        (
            'system',
            system_message
        ),
        (
           'user',
            instructions
        )
    ]
)

# Input either into a .env file or directly into this cell your API key from␣
 ↪serpapi
# Where to get your API key: https://serpapi.com/
tools = load_tools(['serpapi'], llm=llm, serper_api_key=serper_api_key)

agent = create_react_agent(llm, tools, prompt)
agent_executor = AgentExecutor(agent=agent, tools=tools, verbose=True,␣
 ↪handle_parsing_errors=True, max_iterations= 4)

# Notice how we only pass in "input" and not anything else. The Agent Executor␣
 ↪automatically produces the list of tools
# and tool names, along with the agent_scratchpad, for us.

executor_output = agent_executor.invoke({'input': 'Who is the president of the␣
 ↪United States?'})

print(executor_output)
print(executor_output['output'])
```

```
Error in StdOutCallbackHandler.on_chain_start callback:
AttributeError("'NoneType' object has no attribute 'get'")
```

```
Question: Who is the president of the United States?
Thought: I don't have enough information to answer the question, as the
president of the United States can change over time. I will utilize the Search
tool to find the current president.
Action: Search
Action Input: Who is the current president of the United States?
{'type': 'organic_result', 'title': 'United States › President
ShareSend feedbackClose menu'}Question: Who is the president of
the United States?
Thought: I don't have enough information to answer the question, as the
president of the United States can change over time. I will utilize the Search
tool to find the current president.
Action: Search
Action Input: Who is the current president of the United States?
{'type': 'organic_result', 'title': 'United States › President
ShareSend feedbackClose menu'}Thought: I need to extract the
relevant information from the search result to answer the question.
Action: Search
Action Input: Who is the current president of the United States? site:gov
['President Biden represented Delaware for 36 years in the US
Senate before becoming the 47th Vice President of the United States.',
'President Biden represented Delaware for 36 years in the US Senate before
becoming the 47th Vice President of the United States.', 'Learn about the duties
of the U.S. president, vice president, and first lady. Find out how to contact
and learn more about current and past …', "President Biden and Vice President
Harris promised to move quickly to deliver results for working families. That's
what they've done.", 'President Biden represented Delaware for 36 years in the
US Senate before becoming the 47th Vice President of the United States.',
'Donald J. Trump is the 45th President of the United States. He believes the
United States has incredible potential and will go on to exceed even its
remarkable …', 'The President of the United States manages the operations of
the Executive branch of Government through Executive orders. After the President
signs an …', 'Kamala D. Harris is the Vice President of the United States of
America. She was elected Vice President after a lifetime of public service.',
'This chronological list contains entries for each president with corresponding
First Spouse and Vice President. Each link brings you to selected images of that
```

```
> Finished chain.
{'input': 'Who is the president of the United States?', 'output': 'Joe Biden'}
Joe Biden
```

We can see here that we're able to perform Google search with SerpAPI to effectively search the internet for current information.

7 What did LangChain save us?

We won't go through this code in-class to save on time, but if you're interested, this is the code that generated the above example using only OpenAI's libraries.

[19]:
```
from openai import OpenAI

serpapi = SerpAPIClient(api_key=serper_api_key)
client = OpenAI(api_key = api_key, base_url = base_url)
conversation = [
{'role': 'system', 'content': system_message},
]
user_input = 'Who is the current president of the United States?'
print("User: " + user_input)
conversation.append({'role': 'user', 'content': instructions.format(tools = 'serpapi', tool_names = 'serpapi', input = user_input, agent_scratchpad = '')})

# Produce up to Action Input. If Action Input is not produced, then the model will continue into a final answer.
thought_and_action = client.chat.completions.create(
    messages = conversation,
    model=model_name,
    temperature=0,
    max_tokens=300,
    stop = ['\nAction Input:'],
)
thought_and_action = thought_and_action.choices[0].message.content

# Repeats tool use until the final answer is created.
while 'Action: serpapi' in thought_and_action or 'Action: Search' in thought_and_action:
    conversation[-1]['content'] += thought_and_action
    action_input = client.chat.completions.create(
        messages = conversation,
        model=model_name,
        temperature=0,
        max_tokens=300,
        stop = ['\nObservation:', '\n'],
```

```python
    )
    action_input = action_input.choices[0].message.content
    conversation[-1]['content'] += action_input
    ind = action_input.index('Action Input:')

    action_input = action_input[ind + 14:]
    observation = str(serpapi.search(action_input))

    # Inject the observation into the conversation.
    conversation[-1]['content'] += "\nObservation: " + observation + "\n" + "Thought:"

    # LLM either wants to search again or has the information it needs.
    thought_and_action = client.chat.completions.create(
        messages = conversation,
        model="Llama-2-70B",
        temperature=0,
        max_tokens=500,
        stop = [],
    )

    thought_and_action = thought_and_action.choices[0].message.content
    if 'Final Answer:' in thought_and_action:
        break

# LLM knows the final answer.
if 'Final Answer:' in thought_and_action:
    print('Full Output: \n\n' + conversation[-1]['content'] + '\n\n')
    # Parse the final answer and print it.
    ind = thought_and_action.index('Final Answer:')
    conversation.append({'role': 'assistant', 'content': thought_and_action})
    print("Llama: " + thought_and_action[ind+14:])
```

```
User: Who is the current president of the United States?
Full Output:



Answer the following questions as best you can. You have access to the following
tools:

serpapi

Use the following format:
Question: <the input question>

Following this, use one of the following options:

Option 1 - You do not have enough information to answer the question:
  Thought: <plans for which tool you will utilize and how you will use it>
  Action: <the action to take, should be one of [serpapi]>
  Action Input: <the input to the action>
  Observation: <the result of the action>

Option 2 - You have enough information to answer the question:
  Thought: I now know the final answer
  Final Answer: <the final answer to the original input question>

For example:
  Question: Who composed the Four Seasons?

  Thought: I should utilize serpapi to search the internet for this information.
  Action: serpapi
  Action Input: Who composed The Four Seasons?
  Observation: Antonio Vivaldi

  Thought: I now know the final answer
  Final Answer: Antonio Vivaldi

In another example:
  Question: What is the chemical formula of water?
  Thought: I now know the final answer
  Final Answer: H2O.

Begin!

Question: Who is the current president of the United States?
Thought:
  Question: Who is the current president of the United States?

Thought: I should utilize serpapi to search the internet for this information.

Action: serpapi
  Action Input: Who is the current president of the United States?
Observation: [{'document_txt': 'Learn about the duties of the U.S. president,
vice president, and first lady. Find out how to contact and learn more about
current and past …', 'link': 'https://www.usa.gov/presidents'},
{'document_txt': 'President Biden represented Delaware for 36 years in the U.S.
Senate before becoming the 47th Vice President of the United States. As
President, Biden will …', 'link':
'https://www.whitehouse.gov/administration/president-biden/'}, {'document_txt':
'The president of the United States (POTUS) is the head of state and head of
government of the United States of America. The president directs the executive
…', 'link': 'https://en.wikipedia.org/wiki/President_of_the_United_States'}]
Thought:
```

```
Llama: The current president of the United States is Joe Biden.
```

[ ]:

```

```