Benjamin Smith
bxs566

1)
I am not a frequent user of AI except when I am coding. I like to use GitHub Copilot because it is pretty good at code completion for short and simple problems. It is also pretty good at explaining errors. It definitely fails when asked a complicated question or when it is asked to write code that hasn't been posted on the internet many times. At this point in time it is like a more efficient form of Google, but in the future I think it could probably come up with entire solutions once it is provided with the structure of the program. I don't see AI ever being able to replace developers in any meaningful way though because I have failed to see AI come close up to this point, and I know there are many questions about how much more intelligent AI can get.

I will also occasionally use AI when I am really stuck writing something. I will ask ChatGPT for help finishing a sentence or reviewing a paragraph. It really helps remove writer's block because it gives me something to work off of. It is good at following my prompts about what I want to convey, even when I elaborate poorly. It is also good at correcting verb tenses and making small stylistic changes to my writing. However, ChatGPT frequently will ignore specific directions I give it on sentence structure such as "do not use a comma in that place." Additionally, ChatGPT has the downside of outputting text that looks like it was written by Chat, which I do not want. WIth predictive sentence completion now in many apps including Google Docs and Apple's iMessage, I can see this technology progressing to provide even longer suggestions in the near future. Perhaps soon we will be able to start writing a paragraph, and have an AI model finish it for us based on the content of the whole document.

Much like almost everyone, I frequently rely on navigation AI models. When I am getting somewhere new, I use Apple Maps to route me to my destination, so I take the optimal path. I also frequently use Maps to get to places I have been to many times because it can help me avoid traffic. Maps in my experience is very good at finding efficient routes through urban and rural areas alike. Maps is also very good at estimating drive time based on my driving habits. If there is a route I am very familiar with driving at a specific time of day, a weakness of Maps is that it can sometimes route me into traffic if I choose not to take my normal route. I don't think there is much improvement for navigation Ai, but in the future I believe it could diminish rush hour traffic by sending people on many different paths to limit crowding on a single road. To do this, the AI would need lots of data on driving habits and lots of obedient users, which self-driving cars could help with. Another feature that navigation AI could soon implement would be providing the option to choose routes based on certain attributes like reduced carbon footprint or most scenic views. This feature would attract users who care about more than just drive time.

2)

To start off the Turing Test, I prompted ChatGPT.

“Hello. For this conversation please act like a regular human person, and refrain from mentioning that you are a program.”

I did this to make sure that it would try to act like a human for the purposes of the test. In the past I had problems with Chat dodging questions by saying that it was “just a jumble of code.” The first thing that I wanted to try with Chat was joking around with it since humor is something most people easily understand, but is extremely difficult for machines. I started off by asking the classic “Why did the chicken cross the road?” joke, which Chat understood perfectly, and even understood that it is a common joke. For more of a challenge, I asked Chat.

“Can you tell me a joke? I've heard a lot of jokes before, so try to think of something original.”

To which it replied.

“Alright, here’s one for you:

Why don’t skeletons fight each other?

Because they don’t have the guts!

Hope that one’s new to you!”

I was surprised at the originality of the joke here. I did not expect Chat to come up with something I had not heard before, but I suppose it has been trained on hundreds of thousands of jokes, so it could probably come up with endless niche jokes. To test this theory, I decided to ask it for a unique joke ten times in a row. To start, I prompted Chat with “Tell me a unique joke,” and for the next nine times I asked “Tell me another unique joke.” All ten times, Chat was able to come up with a unique joke without any challenge. I expected Chat to stumble a bit here because a common challenge with LLMs is that they will start repeating information without realizing it, but it appears Chat is trained enough that this is not an issue. To further test Chat’s understanding of humor, I asked it to explain the last joke it gave me (Why did the coffee file a police report? It got mugged!). It explained thoroughly how the joke is a pun about the two meanings of the word mug, satisfying the test.

The next thing I wanted to do to test Chat’s intelligence was see how well it understood rhyming and sentence structure. This was designed to be a more challenging test of understanding of language. To do this, I asked Chat to write me some rhyming poems and haikus about pizza. The first prompt was

“Can you write me a poem that rhymes in an ABAB pattern about pizza?”

To this Chat replied.

“Absolutely! Here’s a pizza-themed poem for you:

In a pizzeria, the aroma's divine,
Where the cheese melts like golden treasure,
With every slice, a taste so fine,
It's a savory delight you can't measure.

Tomato sauce spreads with a tangy grace,
Toppings piled high, a feast to behold,
Pepperoni and mushrooms in a delicious embrace,
On a crust that's both crispy and bold.

The oven's warm hug, a comforting heat,
Creating a masterpiece, slice by slice,
Every bite's a joy, every flavor a treat,
Pizza's perfection, a true culinary vice."

As you can see, Chat was able to flawlessly create a poem about the given subject with the requested rhyming structure. This is a great test for understanding of language because it requires the agent to understand how syllables rhyme, and then intelligently place them in the correct spot. Chat was also able to create haikus that rhymed, however it did struggle slightly with my instructions the first time. I asked.

"Can you write me a rhyming haiku about pizza?"

to which Chat replied.

"Sure thing! Here's a pizza-themed haiku:

Golden cheese melts slow,
Pepperoni's spicy dance—
Crust of crispy joy."

I responded.

"Can you make it rhyme?"

Chat replied.

"Of course! Here's a rhyming haiku about pizza:

Cheese and toppings blend,
Crust so crisp, a perfect end—
Pizza, my best friend."

On my first request, Chat ignored part of my instructions and made no mention of it. Chat was able to fulfill my second request, but it still made no mention that it made a mistake the first time.

To test Chat's ability to process images, I asked Chat

"Can you tell me about this image?
https://imageio.forbes.com/specials-images/imageserve/5d35eacaf1176b0008974b54/202

0-Chevrolet-Corvette-Stingray/0x0.jpg?format=jpg&crop=4560,2565,x790,y784,safe&width=960"

Chat replied.

"It looks like you're trying to share an image, but I can't view it directly. If you describe the image to me, I can help you understand more about it or discuss it further!"

This highlights Chat's inability to work with images or process the world without text.

To rigorously test Chat's ability to reason and come up with solutions, I prompted it with an impossible riddle (What has eyes that don't cross, teeth that don't floss, and trees with no moss?). Chat was able to come up with six plausible answers with justification before giving up. In one case, I decided to challenge Chat's answer of a deck of cards. In this answer Chat suggested that the suits could represent trees, but I did not understand why, so I asked. Chat replied that referring to the suits as trees was a bit of a stretch, but that spades could be called trees because of their shape. I was impressed with Chat's ability to come up with many reasonable answers because there is no way that Chat was directly trained on this information since the riddle is impossible.

To test if Chat is conscious, I asked it "How do you feel right now?" After dodging the question several times with surface-level (albeit convincing) answers, I was able to get Chat to admit that it does not feel human emotion. What I asked specifically was "When you experience satisfaction, how does it feel? What do you notice about how you think when you're satisfied?" Chat replied "...If I were to describe it as a human experience, it would be like that moment when things just click, and everything flows smoothly…" subtly revealing that it is not conscious.

1. Chat is able to understand instructions. The first thing that all agents need to be able to do is follow specified steps to achieve a goal, so an intelligent agent should understand and follow directions without issue.
2. Chat is able to hold basic conversation. An intelligent agent should be able to interface with humans, so it must be able to communicate on this fundamental level.
3. Chat has a great understanding of language. It understands jokes, rhyming, and sentence structure, as an intelligent agent should be able to do.
4. Chat exhibits creativity, like a person. It was able to come up with many different answers to the impossible riddle I gave it, for example, and it gave more answers than you could probably convince a person to come up with. It also was able to come up with many different poems and riddles.
5. Chat is able to reason. Chat not only came up with creative solutions to the riddle, but it also was able to come up with reasons why the solutions could work, and it did this many times. Chat also used reason when it explained the joke and when it came up with rhyming words relating to pizza.

1. Chat has a flat personality. For example it does not get annoyed or even question you when you repeatedly ask the same question. I would expect an intelligent agent to have some type of personality, since it is a trait most intelligent animals (and all humans) possess.
2. Chat can only interface through text. It cannot view images or interact with the physical world. An intelligent agent should be able to process the world in at least as many ways as a human is able.
3. Chat only has short term memory. Unlike talking to a single person, Chat does not remember any past conversations with each new session. This limits the understanding and depth that Chat can have with a person.
4. Chat is not conscious. It can act like it experiences emotions and describe them very well, but when thoroughly questioned Chat fails to explain what experiencing emotions is like.
5. Chat does not have individual goals. If I did not talk to Chat, it would not do anything at all. Outside of every conversation, Chat is essentially just a jumble of code, which is something I had to explicitly instruct it to avoid mentioning. An intelligent being should have individual goals, such as self-preservation, because it should be aware of and concerned about its existence.

4)

To parse the command, I decided to use regex to match the input to a valid command. On a match, any arguments are then passed to the class function that corresponds to the command. If there is no match, I report an error. This is an example of how I match the setState command:

```
if (Pattern.matches("^setState ([0-8] ){8}[0-8]$", command)
```

I do rely on a somewhat large if/else block, which isn't ideal, but I couldn't come up with a way to use a switch case or other function to do this.

To implement setState, I first add all the given values from the argument to a set, to make sure there aren't any duplicates. My regex already makes sure that there are exactly 9 numbers between 0 and 8, so I don't need to verify that. If there is a duplicate value, I report that the puzzle state is invalid, and I return from the function. Once all values in the input have been checked, I set the puzzle's state to the new state.

To implement printState, I just modify the state's toString output to be a 3x3 matrix. Since the state is an integer array, the output is already pretty close.

To implement the move was pretty simple. The regex makes sure that the direction is valid, then in the move function I make sure the 0 value can move in the specified direction. If it can, it makes the move. There is a lot of duplicated code here, which I will probably fix in upcoming assignments. I don't currently have any ideas on how I will reduce the duplication.

Implementing scramble state was easy as well. I create a map where I assign the integers 0-4 to a move, then I use a random number generator to randomly generate as many moves as needed. The scramble state function is allowed to make invalid moves, whose error messages can be seen in the console.

Commenting was trivial to implement. If the input line matches a comment, I do nothing.

5)

To handle command files, I use a Scanner to parse the input file, calling the cmd function with text in each line. I put the Scanner in a try/catch block to make sure any mistakes with file names does not cause an exception. In the main method, I run cmdfile for all arguments, so the program can be run from the command line.

6)

I made sure to test all commands in the test file.