a2_troll_reply_bot

February 3, 2024

1 Programming Assignment 2: Trolling a Reply Bot

In this assignment, your job will be to make modify the reply bot below and make your own version of a reply bot (look for the TODO section). Then after you've made it, see if you can troll it. At the end, you will have some reflection questions to answer.
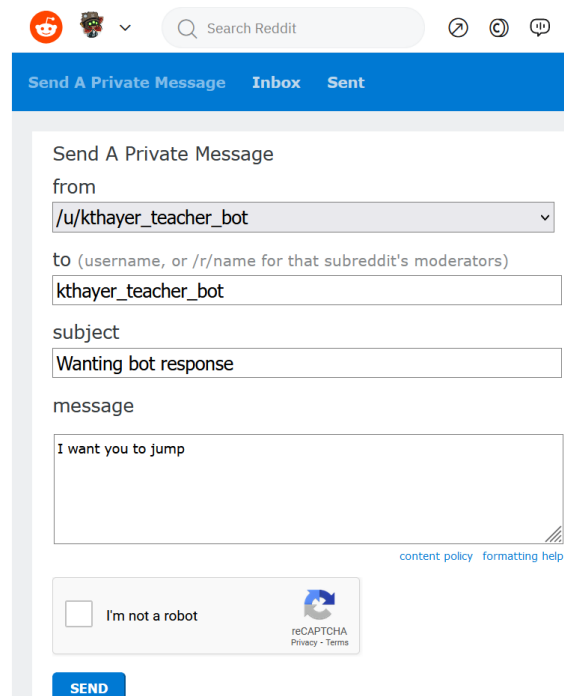
When you are done, you will need to download this file (file menu -> download) and turn it in on Canvas.

Below is the code for the first reply bot from the book, where if you message it: - Subject: "Wanting bot response", body: "I want you to *" (where the* is some action) - then the bot will reply,"I will now _____" (where the ____ is that same action).

1.1 Sending ourselves a message

In order to send ourselves a message we can reply to, go to: - https://www.reddit.com/message/compose/

Then compose a message to your own account with the subject: - Wanting bot response and a message body of something like: - I want you to jump

1.2 Reply Bot

First we need to do our Reddit PRAW setup:

```
[11]: import praw
```

(optional) make a fake praw connection with the fake_praw library

For testing purposes, we've added this line of code, which loads a fake version of praw, so it wont actually connect to reddit. **If you want to try to actually connect to reddit, don't run this line of code.**

```
[10]: %run fake_apis/fake_praw.ipynb
```

```
<IPython.core.display.HTML object>
```

To use this on your real Reddit account, copy your developer access passwords into the reddit_keys.py file.

```
[12]: import reddit_keys
```

```
[13]: # Give the praw code your reddit account info so
      # it can perform reddit actions
      reddit = praw.Reddit(
          username=reddit_keys.username, password=reddit_keys.password,
          client_id=reddit_keys.client_id, client_secret=reddit_keys.client_secret,
          user_agent="a custom python script for learning for " + reddit_keys.username
      )
```

1.2.1 find my latest message

We need to find our latest message in our inbox

We do this by looking in our reddit inbox for messages (we limit it to one, since we just want the latest).

It doesn't directly give us the one message (instead it is in something called an "iterator"), but we can use the `next` function to get the message out.

We then display the subject of the message just so we can see that it found something.

```
[14]: # Look up the subreddit "cuteanimals", then find the "hot" list, getting up to␣
      ↪1 submission
      messages = reddit.inbox.messages(limit=1)

      # get the first submission off the list (should only be one anyway)
      latest_message = next(messages)

      # display the subject and body of the message, so we can see what we found
      display("latest message subject: " + str(latest_message.subject))
      display("latest message body: " + str(latest_message.body))
```

```
'latest message subject: Wanting bot response'
```

```
'latest message body: I want you to jump'
```

1.3   TODO: Modify this section (If message matches our pattern, reply)

You must must modify at least one thing in the code below. You could change the expected patterns, or you could change the code that replies, or you could change both.

We will now see if the message matches our pattern of a message subject of "Wanting bot response" with a message body of "I want you to _____" and then we will reply.

First we will create strings with the patterns we are looking for and save them into variables.

```
[15]: expected_subject = "Requesting bot assistance"
      expected_body_pattern = "Please perform the following task: "
```

We will check if the message has the subject we are expecting. If it does it will check if the essage body starts with the expected pattern. If it does, then we will find the action from the end of the message body text (based on the expected_pattern length), and reply using that action.

We also add "else" cases for when we didn't match the patter, and display a message saying what didn't match.

```
[17]: # Check if the message subject matches the expected subject
      if latest_message.subject == expected_subject:

          # Check if the message body starts with the expected pattern
          if latest_message.body.startswith(expected_body_pattern):
              # Extract the action from the message body text
              action = latest_message.body[len(expected_body_pattern):]

              # Create a reply message indicating the action
              message = "Acknowledged. Initiating execution of task: " + action

              # Send the reply message
              latest_message.reply(message)

          else: # Code executed if the message body doesn't match the pattern
              display("The message body (" + latest_message.body + ") didn't start␣
      ↪with our expected pattern: '" + expected_body_pattern + "'")

      else: # Code executed if the message subject doesn't match the expected subject
          display("The message subject (" + latest_message.subject + ") didn't match␣
      ↪our expected subject: '" + expected_subject + "'" )
```

```
"The message subject (Wanting bot response) didn't match our expected subject:␣
↪'Requesting bot assistance'"
```

1.4 Reflection questions

1. What changes did you make to the reply bot?

The changes made to the reply bot include modifying the expected subject to "Requesting bot assistance" and updating the expected body pattern to "Please perform the following task:". Additionally, the acknowledgment message was adjusted to "Acknowledged. Initiating execution of task:" followed by the extracted action.

2. How could you troll this bot? Give an example of a message that would let you troll it. Or, if you don't think it can be trolled, explain why.

One way to troll this bot would be to send a message with a nonsensical or absurd task, such as "Please perform the following task: Dance like a pineapple on a skateboard." This message would trick the bot into acknowledging and attempting to execute a ridiculous task, revealing its lack of contextual understanding.

3. What limitations does trying to prevent trolling put on your ability to create a bot? (write at least 3 sentences)

Attempting to prevent trolling puts limitations on the bot's flexibility and adaptability in responding to user requests. It may require implementing stringent pattern matching, which could result in false positives and filtering out legitimate requests. Additionally, overzealous measures to prevent trolling may stifle creative or unconventional interactions, limiting the bot's effectiveness in engaging users.

4. Pick two ethics frameworks and compare how they might evaluate the responsibility of someone who is creating a reply bot? (write at least 6 sentences total)

Virtue ethics would evaluate the responsibility of someone creating a reply bot based on the character traits demonstrated in the design and implementation process. A virtuous creator would prioritize honesty, transparency, and the promotion of user well-being, ensuring the bot's responses align with ethical principles. Conversely, an unethical creator may prioritize efficiency or profitability over user welfare, leading to the development of a manipulative or deceptive bot. But feminist ethics would emphasize the importance of inclusivity, diversity, and empowerment in bot creation. A responsible creator would consider the potential impact of the bot on marginalized communities and strive to mitigate biases and discrimination in its design. This framework would critique bot creators who perpetuate stereotypes or reinforce existing power imbalances through their creations, urging them to prioritize social justice and equality.

5. Pick two ethics frameworks and compare how they might evaluate the responsibility of someone who is considering trolling a reply bot? (write at least 6 sentences total)

Virtue ethics would approach the consideration of trolling a reply bot by focusing on the character traits and moral dispositions of the individual contemplating such actions. A virtuous person would refrain from trolling as it contradicts virtues such as honesty, empathy, and respect for others' autonomy. Trolling undermines the goal of fostering meaningful interactions and mutual respect in online communities, reflecting poorly on the character of the individual engaging in such behavior. But Rights ethics would evaluate the responsibility of someone considering trolling a reply bot by emphasizing the rights and freedoms of both users and creators. Trolling violates users' rights to engage in online interactions without harassment or disruption, infringing upon their autonomy and dignity. Additionally, creators have a right to develop and maintain bots without interference

or malicious intent from others, making trolling an unethical violation of their rights and freedoms.