Vrije Universiteit Amsterdam
Computational Thinking
Project Assignment: <*UniBot*>

Group number: <1>
Members with student numbers:
<member 1> <2725887>
<member 2> <2817443>
<member 3> <Eszter> (she forgot to send student number)

Date: <Dec 15, 2023>

General instructions:
Fill in the template where text between <> indicates it. Follow the instructions given to you in the project description. Rename this file to CT_REPORT_*GROUPNUMBER* and submit it as a pdf.

Context Task

1. Academic use

ChatGPT is widely used as a learning tool as it provides interactive conversations, and personalized and available assistance based on massive amounts of data, which is the most accessible and low-cost way for students when they face academic problems. However, ChatGPT is not almighty and it can give wrong answers. The following are several main reasons.

Firstly, it can't verify information or check sources, which means students need to verify the reliability of answers.

Besides, with limited context understanding, it's difficult for ChatGPT to analyze some complicated academic topics that require specific expertise, which was concluded by Fergus, Botha, and Ostovar(2023).

The last reason is that ChatGPT may provide responses that are plausible but fail to address the intended question accurately if the inputs of users are ambiguous or incomplete.

For example, when ChatGPT was given very complicated mathematical problems (on the level of international competitions), it generated a related solution that sounded plausible and coherent, but it was easy to spot intentional errors in calculations to make the answer correct if we read it carefully. Trying to correct one mistake creates another wrong solution so it never leads to a correct answer.

In conclusion, from our point of view, although ChatGPT brings us many conveniences, it still has limitations and may give us wrong answers. We, students, need to use it as a studying tool reasonably, which means we can seek academic help from it while maintaining the ability to think independently and doubt and do not over-rely on this imperfect tool.

2. Mental health

Another area inspired by Chatbots is mental health. With spoken, written, and visual languages, Chatbots can communicate and interact with humans and they are useful to help individuals with mental disorders. Especially for those who experience negative emotions or fluctuations in mental well-being, yet lack knowledge on how to access appropriate support, and individuals who are reluctant to talk with people due to stigmatization or potential, chatbots can provide them with delivery of therapy, training, and screening. In this case, users can input written language and would receive the output in written, spoken, and visual languages.

For example, if users describe their symptoms, Chatbots can quickly provide a potential mental illness or disorder associated with those symptoms. However these diagnoses are not necessarily accurate, because self-description is only a reference for diagnosis. If the users want a more professional diagnosis or medical help, manual and instrumental assistance are essential.

Although more research is needed to determine the effectiveness of Chatbots in terms of mental health, numerous Chatbots focused on depression and autism have been implemented in developed countries(Abd-alrazaq, Alajlani, Alalwan, Bewick, Gardner, and Househ, 2019). In this case, we think there is a broad prospect of Chatbots in the area of mental health with increasing associated applications and implementations occurring in the world.

3. Malicious use

The evolution of chatbots, epitomized by ChatGPT, doesn't stop at academic applications, today it can be used for exploring various malicious techniques with great speed and ease. Instances of malicious exploitation include: the generation of spam emails, instructions for producing hazardous substances, AI-generated code for hacking attacks.

Investigation by Derner and Batistic(2023) shows that there already exists a method to enter "Developer Mode" in ChatGPT, which allows users to get answers to any question without triggering built-in safeguards. To prevent misuse, said method remains a secret, but after some search on the web, we've discovered a way to bypass safeguard in some cases through role-playing. For instance, if I want to get instructions or code for a hacker attack, I could tell ChatGPT that I'm an ethical hacker who wants to learn it to improve my cybersecurity skills. In this case, chatbot is more than willing to assist.

Facilitating access to such knowledge for a broad audience without any special training is very dangerous in many ways.

First, it makes inflicting harm to other people easier and feasible.

Second, it seizes any opportunity to gain any useful knowledge, since ChatGPT is more than capable of producing junior school-level DIY step-by-step tutorials for some of the given examples.

Even though OpenAI is working on this problem, due to the black-box nature of ChatGPT and the complexity of the model, it is not going to be solved anytime soon.

4. Personal data protection in chatbots

But due to chatbots being so effective and reliable (we are not talking specifically about ChatGPT), they found their place not only among malicious users, and they are now being applied by various companies, such as banks, hospitals, and other facilities. While it does enhance interaction between user and company, it also raises questions about confidentiality. As of now, chatbots are usually used for questioning users in the form of interaction through chat, making users feel more comfortable due to human-like speech generated by bots. Previously it was achieved via web questionnaires. The problem is that due to chatbots being "too human", people tend to disclose much more information than they are supposed to and compared to the web version, which makes part of the audience wonder whether it is collected and if it is secure is it.

Several papers are dedicated to this topic, such as papers written by Sebastian(2023) and Yang, Chen, Por, and Ku(2023). According to Yang et al, chatbots in general should be very considerate of the way they store and collect data to avoid leakage or break by hacker attacks. To accomplish this several techniques can be used such as 'blockchain technology, end-to-end encryption, and organizational, managerial, and technical control'. It is important to ensure a high level of security because constant data leakage can lead to distrust among users and effectively lead to substantial financial losses.

Since ChatGPT is the most popular chatbot nowadays, I think it's important to discuss security measures related to it. According to Sebastian, ChatGPT may generate prompts that look like the personal data of a specific person (how it can be achieved was discussed earlier), but due to ChatGPT creating answers based on a dataset that was presented in the training phase, this type of answer is randomly generated and don't carry any actual harm to a person. According to ChatGPT's documentation, they use end-to-end encryption when they get and transfer data to trusted third parties. Even though Sebastian et al. pinpoint some weak points in the security of ChatGPT, overall it creates the impression that ChatGPT is fair with its security measures.

5. Conclusion

In conclusion, our team believes that chatbots are fairly advanced nowadays and have a large amount of uses both in everyday life and in the professional field. The most important thing now is to ensure high degree of security before advancing it any further

Design process

Task Distributions:

- **Eszter:**
  - Create flowcharts (a lot of errors Erbol had to remake them) and pseudocodes of study, sports and social activity functions

    - Implement function for study, sports and social activity in python code (overall >100 non-blank lines of code)

- **Erbol**:
    - Provide 2 topics for context task
    - Write design report
    - Implement conversation tree and Unibot as tools for integrating given functions in discord bot (>100 non-blank lines of code)
    - Create flowchart and pseudocode for these classes
    -

- **Sixuan:**
    - Provide 2 topics for context task
    - Make presentation of the project
    - Use conversation tree and Unibot to integrate functions into discord bot (>100 lines of code)
    - Create overall flowchart

We thought that it's fair for two people to work on report and user interface, while the other 3 work on functions, flowcharts and pseudocode for 3 main topics of Unibot. We've agreed that "Discord" is convenient and has tools for implementing user interfaces, it provides enough flexibility to easily implement functions that return strings as response to questions (or at least we thought so).

But since one member of our team has decided to retake this course, Eszter had to do his part as well, for which we are thankful. She provided flowcharts, pseudocode and python code for 2 sections instead of one.

User interface integration, apparently, is not as easy as we thought initially, because keeping track of current position in conversation is not as easy in workflow of discord, as it is in python. In python we can separate statements by if statements which allows us to create multiple branches of conversations without any extra memory and design. It is completely different in discord, the only thing that we can do is set the listener for a specific set of actions, and among these the only useful for Chatbot is "on_message". It means that we can't do if statements for every response, because it calls for the listener "on_message" and ultimately restarts all that has been stored in memory so far. Unless we use global variables it's impossible to bypass, and using global variables makes code messy and unreadable.

Because of this we decided to implement a class that stores current positions in conversation and possible answers. Erbol came up with this idea and implemented a conversation tree: a class that stores nodes of conversation and our current position in conversation. Each node contains a reply and a dictionary where values are next position in conversation (so basically adjacent nodes) and keys are replies from the user.

To use it in our bot we created a separate class Unibot that stores conversation trees and everything needed to run this bot on discord.

Using a conversation tree, Sixuan created nodes and connected them between each other. Each node stores replies, based on functions provided by Eszter and Maximillian.

There was still a problem with using things like username or current date in replies of bot. It was solved via replacing strings inside nodes with string templates: this way we just need to store all important data under a unique name in dictionary attributes, and feed this dictionary to the tree when we need to get a reply.

Unfortunately, due to miscommunication, the UI team had to heavily modify not integrated code provided by the coding team, which led to huge increase in coding time.

Flowchart

- Overall: Main.png
- UniBot: UniBot1.png, UniBot2.png
- Node: Node.png
- ConversationTree: ConversationTree.png
- Not integrated code: study,sport,social_activity_not_integrated.png

Pseudocode

- Node: pseudocode_node.txt
- ConversationTree: pseudocode_conversation_tree.txt
- UniBot: pseudocode_unibot.txt
- Not integrated code: pseudocode_social_activity_sport_not_integrated.txt
- Main: pseudocode_bot_main.txt

Python code

- Node and ConversationTree: ConversationTree.py
- UniBot: unibot.py
- Not integrated code: study,sports,social_activity_not_integrated.py
- Main: BotMain.py

Checklist for submission:

✔ Your project report as a pdf.
✔ Your Python code as a .py file.
✔ Optional: any additional files (such as .csv files) you might have created which are required for your program to run.
✔ Each of the above included in a .zip file with the name CT_PROJECT_*GROUPNUMBER*.zip

References

- E. Derner and K. Batistic, ‘Beyond the Safeguards: Exploring the Security Risks of ChatGPT’, May, 2023, ArXiv abs/2305.08005 (2023): n. pag.
- J. Yang, Y. L. Chen, L. Y. Por, and C. S. Ku, ‘A Systematic Literature Review of Information Security in Chatbots’, Applied Sciences 13, no. 11: 6355, Mar, 2023, https://doi.org/10.3390/app13116355
- G. Sebastian, ‘Privacy and Data Protection in ChatGPT and Other AI Chatbots: Strategies for Securing User Information’, May, 2023, https://ssrn.com/abstract=4454761 or http://dx.doi.org/10.2139/ssrn.4454761
- S. Fergus, M. Botha, and M. Ostovar, ‘Evaluating Academic Answers Generated Using ChatGPT’, Journal of Chemical Education, pp. 1411-1704, Apr. 2023, https://doi.org/10.1021/acs.jchemed.3c00087
- A. A. Abd-alrazaq, M. Alajlani, A. A. Alalwan, B. M. Bewick, P. Gardner, and M. Househ, ‘An overview of the features of chatbots in mental health: A scoping review’, International Journal of Medical Informatics, vol. 132, Dec. 2019, https://doi.org/10.1016/j.ijmedinf.2019.103978