

3 Sample Dialogue Sessions

Table 1 shows an example of an interaction with pyEliza. Since this implementation is not web-based but command-line based the user had to install the program from the pypi repository⁴ and running it with the following commands:

Install: `pip install pyElizaChatbotClient --upgrade`
 Run: `python -m pyElizaChatbotClient -a`

Table 1: Sample Session with PyEliza

1	pyEliza: Hello, my name is Eliza. Please tell me your name. User: Hi, my name is John
2	pyEliza: I am not sure if we have already met John, but anyway... What's up? User: No, this is the first time I meet you. I'm fine. What about you?
3	pyEliza: Please consider whether you can answer your own question. User: I'm fine
4	pyEliza: How does being fine make you feel? User: I feel good I guess. Some days better than others but in general happy
5	pyEliza: Do you often feel good you guess. Some days better than others but in general happy? User: Yes, today for instance I have good news at work
6	pyEliza: OK, but can you elaborate a bit? User: yes, my boss congratulates me because one of our clients accepted our offer
7	pyEliza: OK, but can you elaborate a bit? User: I have been contacting them for a while and pursuing them to change to our company as investor advisers
8	pyEliza: Why do you tell me that you've been contacting them for a while and pursuing them to change to our company as investor advisers? User: because, after all this work, finally they accepted and now we will have a good client in our portfolio
9	pyEliza: How does that make you feel? User: very happy and proud
10	pyEliza: Very interesting User: Yes. May be this could mean that I will get a promotion too
11	pyEliza: You seem quite sure. User: Not really, but this is a good success for all of us
12	pyEliza: Very interesting. User: Thanks. Ok. I need to leave. Bye

⁴ <https://pypi.python.org/pypi/pyElizaChatbotClient>

4 Additional capabilities

The current implementation easily allows programmers to modify the set of default answers and regular expressions (see the example below) to allow new interactions. In addition, it includes a function to automatically extract the user's name requested in the first turn.

```
[r'Can I ([^?]*\)??', # Regular expression
 [ "Perhaps you don't want to %1.", # Answers
   "Do you want to be able to %1?",
   "If you could %1, would you?"]],
```

On the other hand, although it is not directly related with the capabilities of the chatbot, our implementation includes: a) the possibility of connecting it as a client to an online platform⁵ by using Websockets and JSON messages, b) saving log files to keep track about all the interactions and debugging messages generated by the client when interacting with the users, and c) the possibility of generating XML files with anonymized turns to allow its posterior annotations and evaluation.

5 Future Work

We plan to extend the chatbot by a) including the possibility of increasingly learning new answers at each dialog turn as done by cleverbot⁶, b) include a history mechanism that will allow the chatbot to answer based on the current and most recent turns, and c) by including a new mechanism based on generative models to allow the creation of new answers instead of the predefined ones following a similar approach as proposed in [3].

References

- [1]. Weizenbaum, Joseph. "ELIZA—a computer program for the study of natural language communication between man and machine." *Communications of the ACM* 9.1 (1966): 36-45.,doi:10.1145/365153.365168
- [2]. Weizenbaum, Joseph. "Computer power and human reason: From judgment to calculation." (1976).
- [3]. Vinyals, Oriol, and Quoc Le. "A neural conversational model." *arXiv preprint arXiv:1506.05869* (2015).

⁵ <http://www.teachabot.com:8000/>

⁶ <http://www.cleverbot.com/>