Brendan Lim

CS4395.001

4/14/2023

Chatbot Report

**System Description**

I made a chatbot for the video game “Old School RuneScape”, also known as OSRS. The chatbot is rules-based like Eliza. It relies on the Old School RuneScape online wiki to look up information that it does not have on hand. The bot is written entirely by scratch (which I am now regretting).

To determine which response the bot should use, I used a very simple frequency test which just determines the ratio of common tokens that the user input and some examples written by me. To avoid high commonalities with filler words, they were filtered out using the NLTK stopwords library.

The chatbot uses a variety of NLP techniques depending on the requirements. The most dominant technique was the NLTK POS tagger, since I needed to determine which words to look up on the OSRS wiki. I used the VADER sentiment analysis to get info from the OSRS wiki that are positive. This was to make a very primitive opinion generator by randomly picking the most positive statement for that item/skill. I tried to also use a sentence parser using SpaCy, however it requires TensorFlow to use properly which might not work on all computers.

**Dialogue Tree**

Users are free to use any of the dialogue within the box.

A picture containing text, document

Description automatically generated

**Dialog Interactions**

**A computer screen capture

Description automatically generated with medium confidence**

**Knowledge Base**

Example Old School RuneScape Wiki Page. Note the formatting due to things like the sidebar and the table.

Text

Description automatically generated

Downloaded page with only text. Sentences would be filtered out to try to remove formatting.

Text

Description automatically generated

**Analysis**

To be honest, there’s a lot of weaknesses to this chatbot. The biggest weakness of the bot is its fragility. Because of its rule-based nature, it’s entirely dependent on the syntax and rules that I specifically program. Additionally, to look up an item in the OSRS wiki you would need to get lucky and type the exact name of the OSRS wiki article. I could not search on the OSRS wiki because that would require a whole new set of document parsing logic.

The OSRS wiki enforces a neutral/informative tone to it, which made it useful if I wanted to pull knowledge out of it, but limited the amount/quality of opinions that the bot could have. Additionally, a large amount of the wiki uses table, plotted graphs, and scripted elements. Depending on if it was an item or an non-playable character or a skill in the game, each article would have its own formatting. As an example, the Bronze Pickaxe has very different formatting when compared to a Bronze Axe due to the differing nature of the tools. As a result, it made a poor resource to lookup information from as a bot, which relies entirely on the wiki page’s text.

I think another issue is that I was expecting my chatbot to be an authoritative source of information. Which is not quite possible with a rules-based chatbot. For example, ELIZA basically worked by mirroring the user, which allowed the user to provide the knowledge and content while the bot just mirrors stuff back. My bot is required to accurately determine what part of the unmarked dataset the user is requesting. If I was doing this again, I would probably choose a better dataset of knowledge to refer to or interface with the ChatGPT to more easily parse what the user’s intentions were.