Cosc 310

Assignment 3

Samuel Miller 179161

Overview:

The purpose of this project is to create a Chat agent that will respond reasonably well to the user input. The specific topic of converastion for the chat agent is to simulate a first date.

The way this program handles the user input is it uses a database of keywords to find responses. The program searches the user input and compares it to each keyword, if a keyword is found then it looks up the matching response and returns it. We also implement a secondary keyword to refine a response that is given to the user.

Instructions:

Import project folder into eclipse, and run the main in \src\main\java\chat\main.java

Limitations:

-The response of the chat program is completely dependent on the size of the database. Currently the database is quite small and therefore doesn't respond to the full extent that it could.

-The input for the games have to be quite restricted right now. The POS tagger is quite restrictive, and not very smart about it's tags. This system actualy works better when given a sentence. The synonym game requires pefect spelling to work.

-does not take into consideration previous input and only looks at the current sentence

Features that can be implemented into an API:

1) The sentance parser & responder: Takes in a sentence and compares them to a database. It then responds accordingly.

2)ChatGUI: A simple gui that allows for a more visualy pleasing experience. Provides the basic needs for a chat program.

3)POS: a simple class that implements the POS-tagger quite smoothly and provides methods to make using the system easier.

4)Synonym Game: a Class that creates and runs a simple game revolving around synonyms.

5)Madlib Game: Provides all the tools needed to make a madlib. Takes in user input and uses a predefined blank mad lib to make a full mad lib. Uses the POS to tag the user input.

New Features:

Gender selection: A pop up will show up when start and ask you to slect a gender. You can select the gender of the chat agent which will load a specific database. This will provide slight variations to each setting, allowing a better experience.

male:

USER: what is your name?

STRANGER: The name is Bond, James Bond. It's nice to meet you, what's your name?

female:

USER: what is your name?

STRANGER: The name is Jesse, Jesse Warren. It's nice to meet you, what's your name?

GUI: Added a visual GUI that makes the program look a little nicer. This will start up imediatly after gender selsction.

Games: I implemented 2 games that will activate when you go off topic for too long.

1) Synonym game: This game gives the user a word, and ask for as many synonyms as the user can think of. This implements the Wordnet toolkit to find the synonyms.

STRANGER: Okay then. So I'll give you a word, and you try to name as many synonyms to the word as you possibly can, sound good?

USER: okay

STRANGER: Okay, how about "retrogrades"

USER: retro

STRANGER: I guess we're both pretty bad at this :)

2) MadLibs: the second game that was implemented was mad libs. It takes in user input and attempts to make a madlib. This implements the POS tagger to check the user input to make the madlib.

STRANGER: Okay then. So how about a mad lib? I actually have one here, They're always worth a laugh.

USER: okay then

STRANGER: Okay so I'm gonna need 1 Verbs, 2 Adverbs, 3 Nouns, 4 Adjectives, 2 Past Tense Verbs.

...