Intro to NLP – Fall 2018 – Mazidi: Chatbot Project

**Project Objective**: Create a chatbot using Python and NLP techniques. The chatbot should be able to carry on a limited conversation in a particular domain using a knowledge base scraped from the web and knowledge it learns from the user.

You may work alone if you prefer, or you can partner with one other person.

Overview: The project has three phases:

1. 40 points: scrape the web for content to build a simple knowledge base

2. 40 points: build a chatbot that can discuss a topic and remember things about the user

3. 20 points: evaluation and report

**Deliverables timeline:**

- September 9: team formation and domain (starter url) submission

- September 30: Phase 1 due: upload web crawler and search code. Be ready to make a 2-3 minute presentation about your topic, the urls you retrieved, and the data you scraped off those websites. Show a sample dialog you would like your chatbot to be able to do.

- November 25: Phase 2 due: upload chatbot code

- December 2: Phase 3 due: upload your report.

- November 28 and later: demo your chatbot to class.

**Part 1 Details**

40 points: web crawler and search techniques.

* + Build a web crawler function that starts with a url representing a topic (a sport, your favorite film, a celebrity, a political issue, etc.) and outputs a list of at least 15 *relevant* urls. The urls can be pages within the original domain but should have a few outside the original domain.
  + Write a function to loop through your urls and and scrape all text off each page. Store each page’s text in its own file.
  + Write a function to clean up the text. You might need to delete newlines and tabs. Extract sentences with NLTK’s sentence tokenizer. Write the sentences for each file to a new file. That is, if you have 15 files in, you have 15 files out.
  + You might need to clean up the cleaned up files manually to delete irrelevant material.
  + Write a function to extract at least 10 important terms from the pages using an importance measure such as term frequency. First, it’s a good idea to lower-case everything, remove stopwords and punctuation. Then build a vocabulary of unique terms. Create a dictionary of unique terms where the key is the token and the value is the count across all documents. Print the top 25-40 terms.
  + Manually determine the top 10 terms based on your domain knowledge.
  + Build a searchable knowledge base of facts that your bot can share related to the 10 terms.

**Part 2 Details**

40 points: build the chatbot

* + Design and build a chatbot. There are a few ways to do this:
    1. example of building a chatbot from scratch - <https://apps.worldwritable.com/tutorials/chatbot/>
    2. AIML - <https://www.chatbots.org/chatbot/a.l.i.c.e/>
    3. ChatScript - <https://sourceforge.net/projects/chatscript/>
    4. find others at chatbots magazine - <https://chatbotsmagazine.com/>
    5. api.ai (now called DialogFlow)
    6. Amazon Lex
    7. Many many other options
  + Maintain a user model within your chatbot system. You should have a different user model saved for each user in your group. The user model should store the user’s name, personal information it gathers from the dialog, and the user’s likes and dislikes. Add personalized remarks from the user model to the dialog engine.

**Part 3 Details**

20 points: Evaluation and report

* + Produce a report (no specific format but you’ll need at least 5 pages) that contains a diagram of your dialog tree, sample dialog interactions, an appendix for the knowledge base you created and an appendix for any user models that were created. Perform evaluations of the chatbot (see J&M Chp. 24, 2nd. ed.) and analysis of its strengths and weaknesses.