Joseph Brown

SHURI.APP : NATRUAL LANGUAGE PROCESS CHAT BOT

1.Introduction

Shuri, is a Natural Language Processing (NLP) chat bot app. She can understand plain English to decipher what you are saying to provide a response. As of right now she can tell jokes, greet you, respond to gratitude, and say goodbye.

2. Design and Implementation

Given NLP was part of the app, I had to think of how the Neural Network is connected, build out a file with some training data, and connect that data with the chat app.

1. Json file that contained the data to process and train
2. A file that used the Natural Language Toolkit and NumPy packages to define specific definitions needed to process the data in the json file.
3. Create a file that trains and creates the Neural Network (modeling file).
4. Training file which puts everything together and trains the data received. This also creates a file with the trained data and saves it to be used in the program
5. The final file is what makes everything work. This is the chat file. Which is the app.

- Packages used:
    - Random
    - Json
    - Torch-pytorch
    - nltk-Natural Language Toolkit

- NumPy
- My own created packages:
  - Train
  - Model
  - nltk_utils

3. Conclusions

This project was excellent! I have been wanting to build out an NLP chatbot for some time and did not think I had the time to learn all the ends and outs of building something like this. I found that there are various resources online to help with understanding of how to build out Neural Networks, how they work, and how to implement them in your current project. The NLP part is the key, since you do not have to have hard coded statements and hope the human communicating with the bot uses those exact words.

The best feature(s) are that you can literally type any message to the but and it will give you a valid response. I love how fast the data trained to gather an understanding of patterns to look for, and that it works seamlessly. Having said this, some of the shortcomings is that the dataset is small there are only 4 tags, where I would like to add way more, such as the weather, options for what the bot can do, the bot also needs a storage database, this way I can possible create an API with the data stored, lastly I did not have enough time to complete the bot to have a GUI, right now everything works in the terminal. All the shortcomings will be implemented in stages, adding more data will be an easier task, automating that data load will be harder. I am looking forward to continuing this project.