Fetch Rewards App Overview

Dependencies

- Developed using Python 3

How to Run This App

You have two choices when running this application.

Option 1: Use Docker to run it (Example on Windows using Docker Desktop)

1. Open Docker Desktop
2. Clone the application from DockerHub
3. Find the image *fetchrewards*
    - Click RUN on the image
    - Specify a port under *Local Host* to run the container on
4. Interact with the endpoints using your favorite tool(s)

Option 2: Fire it up using a command line and Python (Example on Windows)

1. Clone the application from GitHub
2. Create and activate a virtual environment in the app directory
    - `cd .\FetchRewards`
    - `python -m venv env`
    - `env\Scripts\activate.bat`
    - `pip install -r requirements.txt`
3. Activate the application
    - `python main.py`
4. Interact with the endpoints using your favorite tool(s)

Endpoints

Base Endpoint

`/`

This endpoint returns a simple HTML landing page related to the application.

Info Endpoint

`/info`

This endpoint returns my decisions to the points made in the assignment. It scrapes data from the HTML page on your S3 bucket and provides my answers as key:value pairs.

This endpoint accepts GET requests.

Example response JSON

```
{
    "Question 1: Do you count punctuation or only words?": "Answer 1: My
application does not count punctuation",
    "Question 2: Which words should matter in the similarity comparison?": "Answer
2: My application removes common stop words"
}
```

Text Similarity Endpoint

`/textsimilarity`

This endpoint accepts key:value pairs from a POST request. It will accept more than two pairs and returns a similarity comparison between all samples that ranges between 0 and 1.

This endpoint accepts POST requests in the format {key:value}.

For example, if you provide three samples (s1,s2,s3), the endpoint returns comparisons between s1 and s2, s1 and s3, and s2 and s3.

Example response JSON

```
{
    "Sample 1 vs Sample 2": "Similarity between two provided samples is:
0.7692307692307693",
    "Sample 1 vs Sample 3": "Similarity between two provided samples is:
0.38028169014084506",
    "Sample 2 vs Sample 3": "Similarity between two provided samples is:
0.3088235294117647"
}
```

Application Decisions

- Question 1: Do you count punctuation or only words?
    - My application does not count punctuation
- Question 2: Which words should matter in the similarity comparison?
    - My application removes common stop words
- Question 3: Do you care about the ordering of words?
    - My application does not care about the ordering of words
- Question 4: What metric do you use to assign a numerical value to the similarity
    - Number of matched words divided by total words in both variables

- Question 5: What type of data structures should be used? (Hint: Dictionaries and lists are particularly helpful data structures that can be leveraged to calculate the similarity of two pieces of text.)
    - Dictionaries, lists, strings, and tuples are employed