Student: Costanza Rodiguez Gavazzi, Agnese Zamboni, Davide Frova

Due date: Wednesday, 27 November 2024, 11:59 PM

1. Project Update Report

1. Overview

Our Information Retrieval project is progressing well. We have divided the work among three team members, each focusing on different aspects of the project.

2. Frontend Design and Implementation

Davide is responsible for the frontend design and implementation. The initial design research and UI mockup creation have been completed using Figma. The next steps involve implementing the design using Next.js. The design aims to be minimalistic and user-friendly. The current prototype contains simple search bar with the iconic search button, the length of the bar is not too short to incentivize the user to type longer queries. A logo representing the project will be placed above the search bar, this will aim to give the user a sense of the project's identity. Under the search bar we have the three pill shaped filter dropdowns, we identified as possible filters: the charity 'theme', the 'location' and the last filter is still to be decided. After entering the query the user will be presented with a list of cards cotaining various informations about the charities, such as the name, the logo and a "google snippet" like description with highlighted keywords. There are also present the 'thumb up' and 'thumb down' buttons to allow the user to give feedback on the results, this will re-run the query with the feedback in mind.
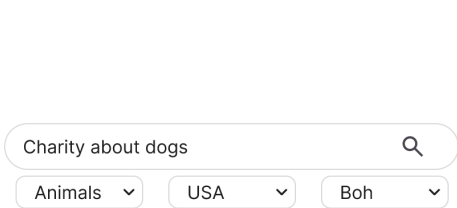Here are some screenshots of the current design:
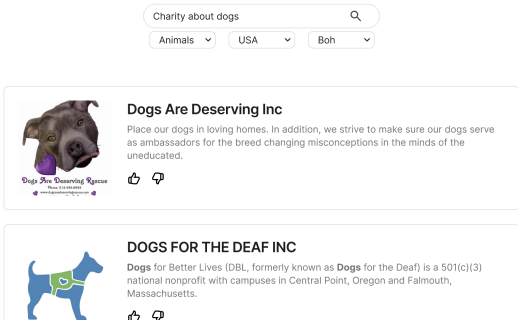

Figure 1: Mockup Search Page

Figure 2: Mockup Results Browsing

3. Data Collection and Processing

Costanza is handling the data collection and processing. She is selecting the three websites that will be used to retrieve the information from and is currently choosing the appropriate scraping techniques and technologies, such as Scrapy and available APIs. The fields to be stored from the charities are being decided.

4. Backend and Indexing/Retrieval

Agnese is focusing on the backend and indexing/retrieval part. We have chosen PyTerrier for indexing and searching the data, and FastAPI to create a simple Python backend that will interface with PyTerrier. The necessary endpoints for the frontend-backend communication are being defined. An draft of the main endpoint comprehends parameters for the user query, the filters and the feedback on a result.

```
GET /search?q=userQuery&filters=filter1,filter2,filter3&feedback={docId:123,score:1}
```