Proposal

Theme 5: Free Topics

What are the names and NetIDs of all your team members? Who is the captain?

The team captain will be **Sahil Gangele**, NetID is `gangele2@illinois.edu`

What is your free topic? Please give a detailed description.

My free topic consists of creating an iOS Application to provide useful search capabilities for an apps App Store reviews.

What is the task?

> The goal of this project is to create an iOS Application that allow's one to intelligently search through an apps 500 most recent iOS App Store Reviews.

An app review is a review a user leaves for a specific app on the Apple App Store.

When searching through a list of reviews, one can take the naive approach of simply showing reviews to users that contain a word in a query. For eg. If a query was “Bad”, then a user would see a list of reviews that contain the word “Bad” in it.

However, by applying NLP techniques (Word embeddings representation, nearest neighbor similarity measure etc.) we can create a search query for “Bad” and reviews that contain the word “Worst” , “Horrible”, “Difficult” will be shown to the end user. This makes the search “intelligent”.

What is your planned approach?

My approach is split into four parts

1. Collect, clean, and format the data of the 500 most recent iOS App Store Reviews for the PBS App
2. Create the user interface to allow a user to search the iOS App Store Reviews and bring up N results. 0 >= N <= 500
3. Use Apple’s NLP tools to create an intelligent search engine that allows a user to query the list of reviews
4. Evaluate system to define “correctness”

Why is it important or interesting?
The App Store is home to one of the only sources of feedback that developers can receive for an App on Apple's Platforms (iOS, iPadOS, MacOS). Therefore, it's an important asset for listening to users' feedback to see what improvements can be made to one's app.

However, with so many reviews given each day, it's hard to gather data for reviews based on a general idea. This bottleneck makes it difficult to communicate these reviews to stakeholders of an App.

To help ease this process, a search engine for App Store Reviews would be helpful in gathering and communicating feedback from users to our stakeholders in order to improve the quality of our app.

What tools, systems or datasets are involved?
The source of the data is from a public API created by Apple: `https://itunes.apple.com/us/rss/customerreviews/page=2/id={APP-ID}/sortby=mostrecent/json` which returns back a list of 50 reviews with a reference to the next 50 reviews. The max number of reviews one can receive is 500 total reviews.

To create the user interface of the iOS Application, Xcode 13 will be used as my Integrated Developer Environment (IDE). `UIKit` / `SwiftUI` will be the libraries used to create the user interface, and the `NaturalLanguage` library will be used to perform various NLP tasks.

What is the expected outcome?
The expected outcome is to have a functional iOS app that at the minimum retrieves relevant reviews to the user given a query search. These results will be ranked according to which review best matches the query. The higher the rank, the higher it is on the list of results first displayed to the user.

How are you going to evaluate your work?
Evaluation is going to be done using the Cranfield Evaluation method.
For example, to test the query "Bad", I will set aside about X reviews and will create relevance judgements on each of them to determine whether a review is "Good" or "Bad". Say out of 50 documents, 15 are bad. When evaluating the system, I'll only run my system on those X number of reviews, and will input the query "Bad". I'll then ensure that of the 50 results surfaced to the user, that the 15 "Bad" results are ranked "high" in the list. In this example, "high" could be the top 25 results. This process can be used for other various

queries, and numbers such as X, and the threshold to determine if the relevant judged documents are “high” in the ranked list can all be adjusted to ensure proper evaluation.

Which programming language do you plan to use?

Swift 5.5

Please justify that the workload of your topic is at least 20*N hours, N being the total number of students in your team.

Based on the planned approach:

1. Collect, clean, format the data of the 500 most recent iOS App Store Reviews for the PBS App
    - This would take 5 hours to complete. The reason parsing this data is difficult is due to all 500 reviews not being present at once. I will have to perform multiple network calls to parse all 500 reviews. This requires threading / asynchronous work to collect all data.
2. Create the user interface to allow a user to search the iOS App Store Reviews and bring up N results. 0 >= N <= 500
    - This will take approximately 10 hours. My plan is to use SwiftUI, as this is a new UI framework introduced by Apple to create simple User Interfaces
3. Use Apple’s NLP tools to create an intelligent search engine that allows a user to query the list of reviews
    - This will take around 10 hours plus to complete. Apple has a video example showing how to use their `NaturalLanguage` library. Learning from this video, applying the knowledge, and ensuring those results display correctly in the app will be the large heavy lifting of the project
4. Evaluate system to define “correctness”
    - This will take about 5 hours as creating a Cranfield Evaluation method will require me to manually judge reviews related to queries to define “correctness”