COMPSCI 2XB3: Computer Science Practice and Experience: Binding Theory to Practice Project Proposal

Project Title: YouRecommend, A Web-Based YouTube Channel Recommendations Application
Lab Section Number: L01
Student Names: Franklin Tian, Jiawei Yu, Kabishan Suvendran, Bowen Zhang
Group Number: 2
Student McMaster Emails: tiany38@mcmaster.ca, yuj130@mcmaster.ca, suvendrk@mcmaster.ca, zhangb82@mcmaster.ca

By virtue of submitting this document, we electronically sign and date that the work being submitted is our own work.

Group Members	Signatures	Date
Franklin Tian		Friday, February 7, 2020
Jiawei Yu		Friday, February 7, 2020
Kabishan Suvendran		Friday, February 7, 2020
Bowen Zhang		Friday, February 7, 2020

Abstract:
Briefly describe the highlights of the proposal (the problem, the solution, the dataset and experiments for verification and validation). (max: 150 words) Hint: you will write the abstract last!

Nowadays, when using YouTube, we are usually recommended many undesired videos and this causes inconvenience. We came up with a solution that provides recommended YouTube channels by directly asking users for their intended channel preferences. Our solution will filter out all of the channels that are irrelevant, since each channel is organized by a category identifier. By choosing the most efficient algorithm, we could solve this problem effectively and efficiently. We used a dataset containing more than 100,000 data entries to support our product. Every piece of data indicates a channel and we could use the statistics of each channel to conduct algorithmic manipulations that will solve the problem. Since the dataset is extremely large, we could extract a small portion of the data for conducting test experiments. By

comparing the recommendations made by our app with YouTube's recommendations, we can determine the accuracy of our product.

1. Objective

In one or two sentences summarize the objective of the project, i.e. what the project will accomplish.

Our idea is to create a web-based application that provides YouTube channel recommendations based on the client's preferences. In addition, it will provide other features, such as a visual representation of the video upload frequency of the recommended channels.

2. Motivation

Clearly and concisely answer the following question: What is the problem that motivates you to build this product? Why is this problem important to be addressed? Who are the users of the product? What are the expected outcomes and the utility of the product?

As a group of individuals who receive their entertainment value and relaxation from YouTube, we have recognized that YouTube's algorithm often suggests irrelevant channel recommendations. Not only is this frustrating, but it also causes users to harbour a strong distaste towards YouTube. Furthermore, it forces users to seek out other media platforms, such as Hulu and Netflix. The nostalgia of growing up watching YouTube makes us view YouTube's faltering reputation amongst the younger audience as a major problem that deserves addressing. As such, we want to create a YouTube channel recommendations web application that provides YouTube users with the best channel recommendations. We expect this application to recommend YouTube channels that conform to the client's viewing preferences, such that the user's love for YouTube is once again renewed.

3. Prior Work

Briefly summarize similar products or solutions to your expected product. Describe any similarities and/or differences. How do you expect to improve the existing products? Make sure to use standard citation style when referring to other work. (max: 200 words)

A similar product that is related to the application we are creating is YouTube itself. From the homepage of YouTube, various recommendations are provided that do not relate to the user's channel tastes or to the videos that the user has previously watched. There are some similarities between our product and YouTube. For instance, they both help users to search for the channel that they desire to watch. However, YouTube's recommended category is completely based on previous channel searches and the videos watched. If multiple users use the same account to watch videos, an inaccurate recommendation result would occur. In order to improve upon YouTube's feature, our product incorporates features to search for recommended channels based on what the user likes to watch at the current moment. Firstly, it asks the user for their channel preference, and then it provides users with options for choosing to sort all the related channels. Next, it recommends all related YouTube channels to users. These channels can be sorted in many ways, such as upload frequency, followers and countries.

4. Input/output and proposed solutions
You should include the following three items separately for this section:

1. **Include the name and URLs of the dataset that you will be using (from Canada or US government open datasets or other sources). Add one sentence for each dataset describing why that dataset is needed.**

   We need a dataset called "YouTube channels ~100000" from Kaggle (https://www.kaggle.com/babikov/youtube-channels-100000) for our "YouRecommend" application because it provides information regarding YouTube channels and their statistics, such as the number of followers (subscribers), the genre of the channel and the channel creation date.

2. **Describe the outputs that your product will be generating in terms of the data and not the interface (e.g., Distance between point A and B, number of cured patients, etc.). For each output add one sentence description of the output.**

   Our product will generate a list of recommended YouTube channels based on the client's preferred genre and this list will be sorted based on the video upload frequency in descending order. The video upload frequency will be calculated and represented visually. The product will also generate a list that is sorted in descending order based on the number of subscribers or the creation date of the YouTube channel.

3. **Briefly describe the proposed solution in terms of how the input will be used to generate the output. You can describe this part as a usage scenario. You may add a figure if it helps to describe how your proposed solution is working. (max 500 words)**

   To begin, our webpage will provide the user with search options for finding their preferred YouTube channels. We would first sort the dataset and then employ binary search to navigate and compare the channels. If there are no matches, then there will be no outputs. Users could also search by the channel category and the amount of subscribers. Our program will first compare the category to each of the YouTube channels and filter out the channels that do not match the category inputted. Then, depending on the size of the remaining channels, we will use the appropriate algorithm to sort them by subscribers in descending order. For example, if there is a large amount of YouTube channels, quicksort will be applied, since it is generally more efficient on sorting large amounts of data entries. If only a few YouTube channels occur, merge sort will do the trick. After searching for the desired category, a directed graph algorithm could be applied to recommend users of other channels that are similar in content and geographical location. Each node/vertex will represent a YouTube channel, while the directed edge will be the country identifier and category identifier. After a particular search, various other channels will be listed along with their statistics.

5. Algorithmic Challenges

Briefly describe which algorithms (sorting, searching, graphs, strings, etc.) you think you will be implementing to solve this problem and what are the challenges that you should be aware of. (max 200 words)

Quicksort will most likely be used in the project because it is the most efficient sorting algorithm for large amounts of data. A searching algorithm that is efficient on large amounts of data is binary search. We could search by the channel category identifier instead of the channel category (String type), since it is an integer value. In terms of graph algorithms, our approach is to use directed graphs. This will provide the user with recommended channels that are related to the channel that they have found. The common challenge we might face is debugging the program for the large dataset. During the testing phase, it would be better to extract a smaller set of data so that debugging could be more time efficient.

6. Project Plan

Include a table that describes at most 7 milestones and their deliverable for your project. Each milestone should have a specific date (e.g., Week 3).

Milestone	Deliverable	Date
Project proposal presentation	Presentation to the class	Week of February 10, 2020
Requirements specifications	Requirements document based on the instructions	March 7, 2020
Project progress checkpoint	The prototype demonstration	Week of March 9, 2020
Final project presentation	Presentation to the class	Week of April 6, 2020
Final project code	The code of the implementation	April 12, 2020
Design specifications	Design document	April 12, 2020
Team peer evaluation	Complete the evaluation form for the other team members	April 12, 2020

References:
You should have at least five references. Use a consistent format (IEEE, APA, etc.) for your references.

Babikov, I. (2018, November 15). *YouTube channels ~100000*. Retrieved from https://www.kaggle.com/babikov/youtube-channels-100000

Graph Data Structure And Algorithms. Retrieved from https://www.geeksforgeeks.org/graph-data-structure-and-algorithms/

Kharkovyna, O. (2019, June 21). *Top 10 Great Sites with Free Data Sets*. Retrieved from https://towardsdatascience.com/top-10-great-sites-with-free-data-sets-581ac8f6334

Sedgewick, R. *Algorithms, 4th Edition*. Retrieved from https://algs4.cs.princeton.edu/home/

Song, R. (2019, August 28). *Building a Movie Recommendation System Using TigerGraph*. Retrieved from https://www.tigergraph.com/2018/07/26/building-a-movie-recommendation-system-with-tigergraph/