LegendaryYT

Project Summary
The current Youtube algorithm is great for recommending videos, but sometimes you want to see trending videos that are more specific to your searches. It is easy to see trending videos on YouTube as it is, but with our project, you can filter trending videos and search through them using your own specific filters and criteria. That way, you can still see popular, relevant videos, but they will be a better match for what you're looking for.

Data Description
We are using the YouTube Trending Video Dataset for the US region. This data is from kaggle.com and was collected using the YouTube API. The attributes being stored are video_id, title, published_at, channelId, channelTitle, categoryId, trending_date, tags, view_count, and likes. To figure out which category goes with each categoryId, we must use the JSON file given.

Basic Web Application Functionality
Users on this web application can search for certain keywords or channels and receive a list of trending videos that match. They can also search for videos that are in a certain category (these categories are predetermined). Finally, they can filter the results by the time period in which they were trending or by when they were uploaded so the users can see recent videos that match what they're looking for or look for videos irrespective of that.

Creative Component
A good creative component of that can improve the functionality of the our application is the implementations of a "Favorites Button" Once a user views the video they have the option to click on a button is clicked

Application Description
This project aims to develop a hot video recommendation application based on the dataset of daily trending YouTube Videos. The application will utilize the data from the USA region to provide recommendations that match users' interests and preferences. The data includes video title, channel title, publish time, tags, views and so on, which will be used to determine the popularity and relevance of the videos.

The goal of this project is to provide users with a seamless experience of discovering new and trending videos on Youtube and to solve the problem of overwhelming choices by

presenting them with personalized and relevant recommendations. The application will consider factors such as the category of the video and the popularity of the video to provide accurate and relevant recommendations. Overall, this project will provide a simple and efficient way for users to discover new and interesting videos.

Usefulness

This application is useful because it allows users to search for popular videos based on filters that they have more control of. Youtube currently has a trending section, but it only shows you the top videos that are trending right now and it doesn't let you search within the trending list or look at old trending videos. The only way to filter the trending videos is to choose one of 3 categories: music, gaming, and movies. Our search function will have more than those 3 categories to choose from and it will let you search the trending list for more specific things.

Realness

Our data contains information about the daily trending YouTube videos. We got this dataset from kaggle.com and it was collected using the YouTube API. This data is updated daily and will allow users to find videos on YouTube they are interested in. The dataset has each entry as a YouTube video with its title name, channel name, publish data, tags that the video is under, amount of views the video holds, and the amount of likes the video holds.

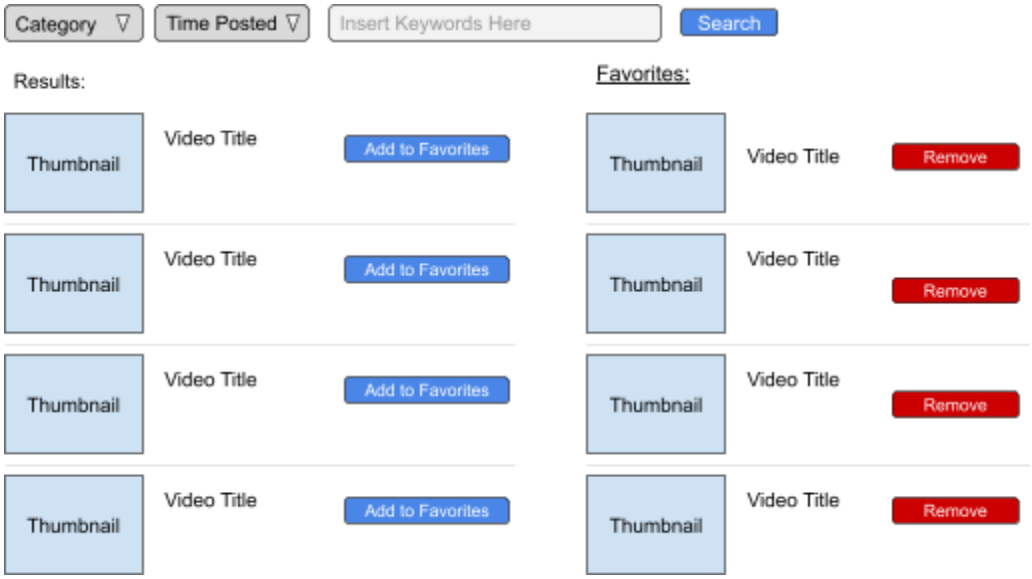
Functionality Description

The main functionalities of the application include:

1. Search: The application will have a search function that allows users to filter the trending videos based on specific keywords, categories and so on. The categories are predetermined and the users can select one of these to see videos that belong to that category.
2. Favorite: In addition to the search function, the application will also have a favorite function which allows users to save videos they like to a personal list. The list can be accessed later to view the saved videos or to delete them.

This mockup provides a general idea for how our website might look. The top is where the user can filter their search by video category, time posted, or a search. The results on the left is where the recommendations will come up. Each result has an Add to Favorites button that allows the user to insert into the database to add the video to their Favorites list (shown on the right). Then the user can update this table by deleting certain videos from favorites, which can be stored as a table on the database.

Low-fidelity UI mockup:



Project work distribution

We will be using **python** to store the dataset information and also to provide recommendations that match the input given by the user which tells us their preferences. Besides, we will use **ReactJS** to deliver the front-end development. We will display the recommendations on the front end which will look like the mockup shown in 9a.

The distribution of tasks and responsibilities for the implementation of the functionalities depend on the team structure and skill set of the individuals.

Below we have listed who will be responsible for the front/back end and their respective languages.

1. Search functionality:
    a. Front-end development: The user interface for the search function and the implementation of the filters and categories can be handled by **Shams Alshabani** and **Yu Zhang**.
    b. Back-end development: The retrieval and processing of the data based on the search criteria can be handled by **Shreya Patel** and **Itay Gozalzani**.
2. Favorite functionality:
    a. Front-end development: The user interface for saving and accessing the favorite videos can be handled by **Shams Alshabani** and **Yu Zhang**.
    b. Back-end development: The storage and retrieval of the data for the favorite videos can be handled by **Shreya Patel** and **Itay Gozalzani**.

The backend systems will be distributed across members based on their expertise and workload, and can be managed using a version control system such as Git to ensure seamless collaboration and maintain accountability for changes.