

Creating Playlists From YouTube API

Shubham Arora-11841080

August 21, 2020

All steps have been explained in the notebook itself.
This file contains description of columns and approach used to make scoring function.

1 Columns Description:

'Topic', 'TimeQueried', 'Video_Title', 'Video_ID', 'Published_At', 'duration', 'viewCount', 'likeCount', 'dislikeCount', 'favoriteCount', 'score': As explained in assignment explanation.

'commentCount': Number of comments on that video

'defaultAudioLanguage': Audio language of video

'Duration in Minutes': Duration of video in minutes

'TimeQueriedSecondObservation': Time when 2nd observation is recorded

'viewCountSecondObservation': View Count when 2nd observation is recorded

'likeCountSecondObservation': Like Count when 2nd observation is recorded

'dislikeCountSecondObservation': Dislike Count when 2nd observation is recorded

'commentCountSecondObservation': Comment Count when 2nd observation is recorded

2 Scoring Function:

Score is computed topic wise and based following methods:

1. If topic we are searching is present in the title of video, then a score of +1 is given to that video.
2. **Duration:** Based on duration different score is given to different videos. As explained in the notebook, duration is divided into three intervals and total number of videos in each interval is computed. Same score is given to all the videos in that interval.

How score of each interval is calculated:

Fraction of videos in taht interval is computed as the score of that interval.

Example 1. *If $videos_in_first_interval$ is the number of videos in first interval and $total_videos$ is total number of videos, then*

$$score = \frac{videos_in_first_interval}{total_videos} \quad (1)$$

3. **Like/Dislike:** In this fraction of likes with total votes (likes+dislikes) is computed and given as the score of that video. Score if given based on following:

- (a) likeRatio ≥ 0.9 , then Score = 2
- (b) likeRatio < 0.9 and score ≥ 0.75 , then Score = 1
- (c) likeRatio < 0.75 and score ≥ 0.5 , then Score = 0
- (d) likeRatio < 0.6 , then Score = -1

4. **Audio Language:** If video audio is not in English, then -1 is deducted

5. **Recent Trends:** This one will be different for all videos as in like/dislike case.

How score from recent trends is computed:

Score is computed as fractional change in the view, like, dislike, comment level.

Example 2. *If $first_views$ is the number of views in first observations, $second_views$ is number of views in secong observation, then*

$$fractionalChange = \frac{second_views - first_views}{first_views} \quad (2)$$

$$score = fractionalChangeInViews + fractionalChangeInComments + (1.5)*fractionalChangeInLikes - 2*fractionalChangeInDislikes$$

6. Based on these scores dataframe is sorted and videos with decreasing scores whose duration is less than or equal to 18 min. and audio language is English is added to playlist.