AI HACKS CHALLENGE

Team: Somaia R. Saba, Teresa Gao, Erin Shin, Melissa Calvert

OUR ALGORITHM

Our recommendation system combines a popularity and collaborative filtering approach. It produces up to 5 recommendations when given a video that has been watched.

```
29 r = recommendation(624)
30 print(r)
```

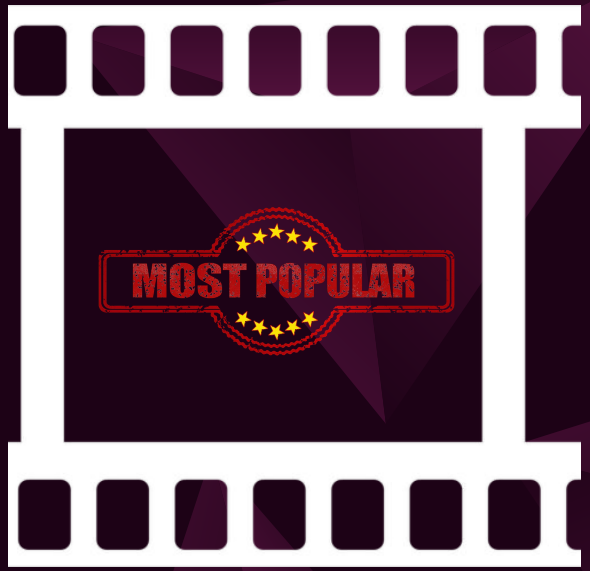
Input video id to function

```
In [1]: runfile('C:/Users/somai/OneDrive/Take 2/
OneDrive/Documents/upenn-ai-hackathon/
recommendation.py', wdir='C:/Users/somai/OneDrive/Take
2/OneDrive/Documents/upenn-ai-hackathon')
[710, 688, 687, 731, 2587]
```

Up to 5 recommendations are in the output

HOW IT WORKS: POPULARITY REC

One of the recommendations will always be the most popular video available. If the video just watched was the most popular video, the next most popular video will be recommended.



HOW IT WORKS: FILTER REC

The other recommendations are computed using a collaborative filtering method. We created a system where each video is a node. Nodes are connected by edges that are weighted through a sum of how many people watch both videos as well as any tags that are common between videos. The views are weighted as $(½)^n$ for each unique patient where n = number of views (if the video has been watched 3 times by one patient and once by another, the edge is weighted as 2.75). Given a video watched, the recommendations would be the videos connected to that node with the highest weights. Up to four of these recommendations are included in the overall recommendation list.

WHY THIS ALGORITHM?

We designed this algorithm because when we look for videos to watch, we usually ask our friends who have a similar watch history if they've watched anything new. We also did not want to get stuck watching videos of only one genre so we included the most popular video as a recommendation as well.

IMPROVEMENTS

We thought of how we could improve the system. If we could have access to video ratings, we would ask the patient to rate each video and use this to assess popularity instead. We would also incorporate this into the collaborative filter aspect so that edges between nodes would be weighted higher if patients who watched both videos rated them highly. Patient watch history could also be incorporated so that videos already watched are not recommended.