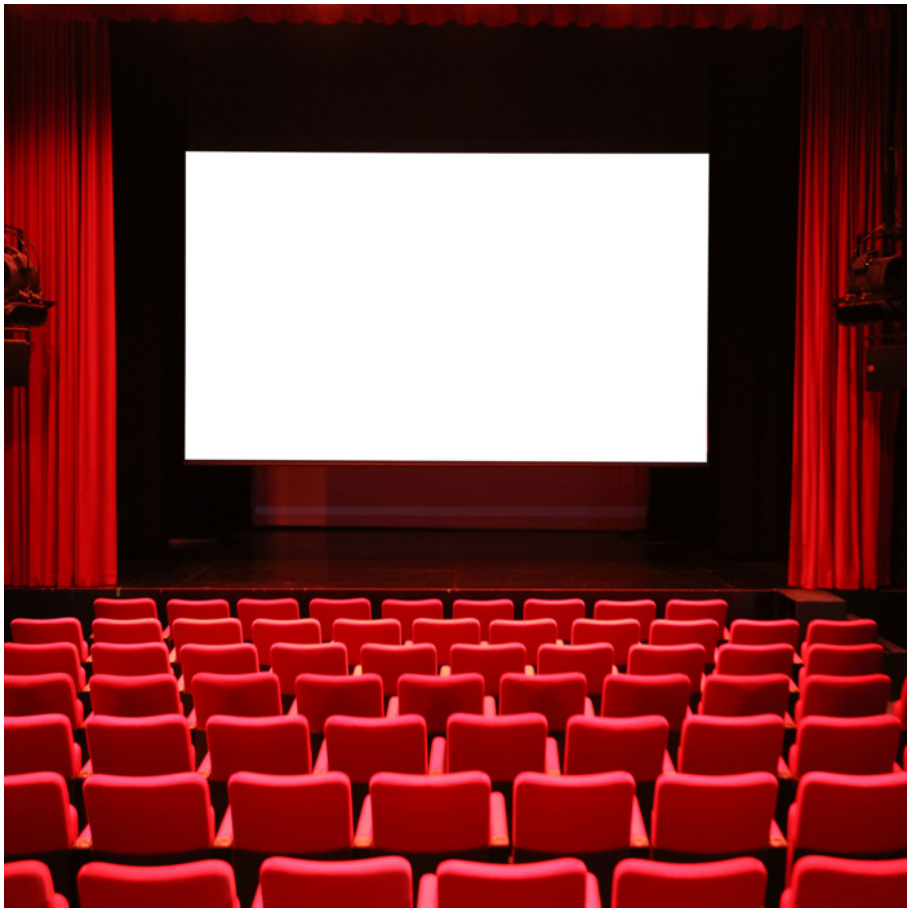

The Big Picture

CS 520 Final Question 5

Spring 2019

Xuenan Wang

NET ID: xw336

THE BIG PICTURE

Pick one of the following. Identify some of the properties and abilities it would need. Identify at least five issues that would be faced in trying to implement or build such a thing, and possible algorithmic solutions for each based on the material covered in class. Credit will be given for thoroughness and detail.

- **MovieAdvertisingAndPromotionBot**

- Properties:

1. IMDB Rating
2. Genres
3. Rated PG
4. Director
5. Year
6. Stars
7. Time Length
8. Language

- Abilities:

1. Collect user's preference of movies
2. Tell if a movie satisfy user's preference
3. Recommend a similar movie
4. Find potential user that may be interested in one movie

- Issues would be faced

1. How to collect users' preference?

Firstly, we ask user to grade some movies he have seen and we use these score as a scale to evaluate how much he likes this movie. After that, we use the score user gave for one movie to multiply the features, which indicates that how much he love this feature. Then we combine all feature scores and from there we can tell some information about user's preference. To make this happen, I think here we can use K-Nearst-Neighbors algorithm to do that. By calculating the relative euclidian distance of each feature, we can tell how much a feature is affecting this user on choosing a movie. For instance, someone could be a huge fan of Harrison Ford, then it's reasonable

to say that the feature STAR will be more influential to him when STAR='Harrison Ford'. After a certain amount of data, we will be able to have a model that can basically tell which kind to movie this user may like. Or we can tell if a new movie will meet his preference or not.

Another idea about collecting user preference is that we can see if this user finished watching the movie. If not, we search our previous database to see if he have seen it before, if this is the first time he watch it and he didn't finish and would not finish it in a month, we probably can say that he didn't like it and refuse to continue watching it. In this case, we can say that this movie will be marked as a negative kind of user's preference. And it can be a big negative contribute to user's preference. Similarly, if user repeatedly watch one movie, we can see it as that this user really.love this movie and we can give it a big positive influence on the user preference.

If we may, we can also collect some other information about user. Like gender, age, nationality and political tendency, etc. These information will gave us power to cross compare the movie preference from users who have similar situations. It's clearly to say that when people having similar background, they are likely to have similar preference. We can use it as a tool to further improve user's preference. To do that, we can use clustering algorithm. Since we don't know exactly how these background are contributing to user's preference, it's reasonable to choose an unsupervised learning algorithm to do that. We choose 100 movies from each user having similar background, and see how many are them are also on other's list. Then we sort those movie by how many time it showed, the more times it showed, this movie is more popular in this certain background. We can mark those movie with these certain background to use it as a bases when trying to advertising.

2. How to classify all the movies?

We classify movies by 8 features, and also mark movie with what background are more likely to enjoy this movie.

3. How to find potential user to send advertisement?

As explained above, users with similar background are likely to have similar taste. We find potential users by searching which user are having the same backgrounds that most people in this background are likely to enjoy this movie. We sort these potential user by how much they are likely to enjoy(the similarity of their background and the preferred background of the movie) and we only advertising this movie to those potential user having relatively high similarity so that users may not be interested in this movie will have a less possibility to receive this advertisement.

We can monitor if there's an important event recently, and if there is, we can search keywords about this event in our movie database so that maybe at this time people will want to watch such a movie. For instance, when the Notre Dame in Paris was burned, people are likely to watch movie about Notre Dame in the memory of its previous appearance. Also this strategy can be applied on certain time, like Christmas or Valentine's Day. This strategy is based on the fact that people have tradition to do something on particular time or when something happens. We can make this come true by marking movies with certain time and event as keywords. And we can use it whenever we need and when we need it, we can simply search the keywords. Of course, we can make this advertising system more efficient by marking whether or not a user will follow this kind of trend. The idea is similar to above.

4. How to find a similar movie?

We already have a huge KNN model that having relation among movies, we can tell if two movie are similar by how much they are close, or to be precise, how much the feature combined and the value are.

5. How do we get the features and datas about one movie?

To begin with, we can get it from internet when all the movie comment are. We can use a LSTM model to classify whether or not a comment is good or bad, and by calculating how many percentage of the comment are good, we can tell how people think about this movie. And a lot of features about one movie is a fact and all we need is to search and find the answer. That should not be a problem.

Bonus: In honor of her making a full recovery,

i.What did my dog dress as for Halloween?

Superman? Or maybe supergirl is more appropriate. I remembered this picture you used when elaborating clustering algorithm.

ii.Draw a picture of my dog in her Halloween costume.

You get the basic idea. Really not good at drawing.