Context Based Filtering

By Prabhdeep Singh and Nathan White

**Motivation and Problem**

The topic of recommender systems is massive and complex. As a result, we are only able to cover topics to a certain depth within the scope of the classroom. Until now we have been working with data if an item was purchased or not. The problem involves recommending music to users who are driving their cars, based on the context they are in. Since driving is something most people do, and many will listen to music in the car, this can provide and improvement for a multitude of people.

**Idea**

At the highest level, there are two parts to solving the problem at hand. Those two parts are research and implementation. Before we can begin implementing any recommender, we need to research the method to understand it from a mathematical sense, as well as understand the advantages and disadvantages of the algorithm from a programming viewpoint. Once we have performed the appropriate research on how to understand and compare the data, we can begin our implementation. We will be implementing the recommender system in Python. An initial idea involves seeing what similar users have liked, based on the context and recommend similar songs.

**Dataset**

Our dataset for this project will be an excel sheet with a list of user who rated certain songs and the context they were in. This context includes items such as the landscape they are in, mood, time of day, etc... There is also a music category associated with each song. This includes categories such as rock, country, jazz, etc… The dataset is about 4000 user item rating from 1 to 5 based on 8 different contexts of about 3 cases each. Each song can also belong to 1 of 10 different music categories. There is also other data like song title, album, and artist that most likely will not be taken into consideration.

**Timeline**

In terms of the timeline for this project, the tentative time breakdown is to research for two weeks, implement for two weeks, and have one week reserved as a “flex” week that we can allocate towards research or implementation.

**Responsibility**

Prabhdeep and Nathan will split the tasks required to complete this assignment fairly. The research portion will be done separately, followed by a collaboration to identify which approach will be better suited for the project. Based on what is found from the research, one person will be responsible for modifying existing projects to match an agreed interface as well as doing more isolated tasks. The other will be responsible for implementing the main algorithm. There will obviously be collaboration if one person is stuck, but this is an attempt to minimize overlap between developers. This is also subject to change based on how the research portion shakes out. R