ITCS 6162 Programming Assignment Report

Emory Soper

Movie recommendation systems are widely used in the age of digital entertainment. To keep users watching content and invested in a streaming service, recommendation systems are used to provide users with more content that they will likely enjoy. For this assignment, three different methods were used. User-based collaborative filtering checks which users are similar to each other using past ratings and gives a user recommendations based on what similar users have rated highly. Item-based collaborative filtering checks if items are similar and can use this to recommend items that are similar to a particular item. Random walks are used in a graph-based approach to organically test which users and items are most closely connected through the visits that nodes obtain on a random walk from a particular starting point.

The MovieLens 100k dataset was used because it contains a large amount of data for users, movies, and ratings. This amount of data can approach the amount of data that a large streaming service might be working with and is a good starting point for testing out recommendation systems. This data was extracted into three csv files. One had a user id along with that user's age, gender, and occupation, although that additional data was not used. The next had a movie id along with the title of that movie. The final one had a user, the movie they rated, the rating, and the time that they rated it. This data was combined together to form a matrix showing which users rated which movies and what the ratings were.

The user-based and item-based collaborative filtering were formed from the rating matrix described above. To calculate similarities, the cosine similarities were taken between the user rows and the movie columns, respectively. For item-based filtering, the recommendations were simply chosen by seeing which movies has the highest similarity to the chosen movie. For user-based filtering, the expected rating for each unranked movie by the user was calculated by using the ratings from other users weighted by the similarity to the primary user. The highest calculated ratings were then recommended.

The random-walk-based Pixie algorithm was based on an adjacency matrix formed between the users and movies represented as nodes. An edge formed between a user and movie if the user had rated the movie. This forms a bipartite graph where a random walk continuously switches between connected users and movie. The recommendations for the user or movie represented by the starting node are given based on which movie nodes were visited the most by the random walk.

With such a large dataset, it is difficult to draw large conclusions based on a small number of tests. The user and item based collaborative filtering appeared to be consistent and provided good recommendations. The large number of connections present within the adjacency matrix graph makes it difficult to find consistency in the random walk recommendations, even with walks as long as 10000 steps. Even so, there was some consistency when testing the same user multiple times as such a large scale. To fully explore these types of recommendation systems, a much larger amount of tests and some sort of automatically calculated accuracy would be needed to test the effectiveness thoroughly. The current implementations are fairly basic compared t what is used by companies, so research into better methodologies and hybrid methods would help to examine the best methods. Recommendation systems are a complex and still growing methodology that has a lot of room for exploration.