**Recommendation Systems**

**Team Members (and the roles of each member)**

**Collaborative Filtering**

Fatma Mokhtar

Vivek Kammula

**Matrix Factorization**

Mark Pop

Henry Zhao

**Unsupervised K-Clusters (Hybrid Recommendation)**

Jonathan Chance

Prakhar Praveen

**Description of the problem you try to address**

Recommendation systems have become more and more popular over the last two decades due to a rise in e-commerce and online media consumption. In recent years we’ve seen an explosion of different media companies creating their own streaming service, which introduces a need for systems that can suggest relevant content to users in a timely manner from millions of options in a company's movie catalog. Because the increasing popularity of this issue in regards to streaming services has been growing at such a rapid rate, we decided to stick specifically to movies in regards to our project. Since this is such a widely researched topic there are many approaches that people have tried to tackle this specific issue. Instead of only choosing one, we have decided to implement several approaches and compare which performs better for both time taken to retrieve the results (efficiency) and the correctness at which a system can recommend a movie (accuracy).

**Preliminary plan (milestones)**

For our project we decided to implement several different machine learning approaches and compare the results. In order to complete the project in a more efficient manner we broke off into three teams of two to implement and test each approach. We plan to work on these different models in parallel and once completed compare the results to see which is best in terms of efficiency and accuracy. A description of the models and method of comparison is listed below.

**Collaborative Filtering**

Collaborative Filtering is a technique used in several recommender systems. It is based on the premise that people generally receive the best recommendations from others with similar interests. It relies on matching people with similar interests and providing recommendations.

In the narrower sense, in collaborative filtering, the predictive model makes predictions for users based on information collected from a large number of other users. It works on the basic assumption that if Alice and Bob both like a movie, then their other interests must match as well. By obtaining a partial list of a user's interests, based on a vast amount of information possessed by the predictive model about other users' interests, the model recommends things to the user. This system is used by Netflix when a new user creates a Netflix account and logs in for the first time.

In the wider sense, collaborative filtering is the process of filtering for information and patterns by collaborating among multiple data sources. Applications that are based on the process of Collaborative Filtering, depend on very large data sets. The process of Collaborative Filtering has been used in several sectors such as mining, the financial world and several e-commerce websites.

Collaborative Filtering algorithms can be divided into 2 classes - memory based algorithms and model based algorithms. Memory based algorithms store the data needed for making predictions within their memory. Model based algorithms create patterns offline which are used to make predictions. Another class of collaborative filtering algorithms are probabilistic algorithms and non probabilistic algorithms. Probabilistic algorithms are dependent on an underlying probabilistic model. Non probabilistic algorithms are used more in general.

Non probabilistic algorithms include 'User based Nearest Neighbor Algorithm' and 'Item Based Nearest Neighbor Algorithm'. User-based algorithms generate a prediction for an item by analyzing ratings for the item from users similar to the user predictions are being made for. Item-based nearest neighbor algorithms are the transpose of the user-based algorithms. While user-based algorithms generate predictions based on similarities between users, item-based algorithms generate predictions based on similarities between items.

Most probabilistic algorithms calculate the probability that, given a user u and a rated item i, the user assigned the item a rating of r: p(r|u,i). A predicted rating is calculated based on either the most probable rating value or the expected value of r. Bayesian network models are the most popular probabilistic algorithms.

Beyond the basic accuracy measures there are several measures for evaluating the performance of a collaborative filtering model. Some of them include- novelty, coverage, learning rate, user satisfaction metrics etc. All of these measures serve as better metrics at evaluating a collaborative filtering model's performance than accuracy.

Challenges of using collaborative filtering models are, like most challenges these days with security and privacy. Since these models rely on collecting information from users, there's always a privacy concern regarding what user information is being collected. In centralized servers there's a risk of the user information getting compromised if the server is compromised. Distributed servers are more protected from such compromises, but even in such systems, there's a risk of user information being transmitted in an unauthorized manner. A potential solution for this is to use encryption keys.

This is a brief summary regarding collaborative filtering, the algorithms used, the metrics that can be used to evaluate the model and the challenges faced by such models.

**Unsupervised K-Clusters**

Hybrid recommendation systems aim to combine existing recommendation system techniques in the hope to gain better performance. We intend to evaluate this technique by reproducing the ideas of Achhab et. al. who combine collaborative filtering techniques with that of unsupervised K-clusters. Given a set of movies, the hybrid recommendation system would first extract features from the movie dataset using techniques such as PCA feature extraction. This is to reduce the amount of genres and thereby reduce the dimensionality for more efficient processing. Once the features are extracted, these are then fed into a K-clustering algorithm which groups the movies into a specific set of clusters.

After each movie is assigned to its cluster classification, for each user, the highest rated movies of each user are compiled if the movie exceeds a certain threshold and then determine which K-clusters the user prefers the most. From this information, a user profile can be created, and then collaborative filtering techniques can be applied to the subset of movies that belong to a given cluster a user is associated with and thereby recommend movies the user is most likely interested in.

**Matrix Factorization**

Matrix factorization was demonstrated to be superior to the classic nearest-neighbors techniques with regards to product recommendations at the Netflix Prize competition. It allowed additional information such as implicit feedback, temporal effects, and confidence levels. With user experience becoming increasingly important, especially for entertainment products, new recommendation techniques rise to the top; this is where matrix factorization comes in.

Matrix factorization works by characterizing items and users by vectors of factors that are inferred from the item rating patterns, with high correspondence between item and user resulting in a recommendation. This provides good scalability along with good predictive accuracy.

**Compare using K-Fold Cross Validation**

Each model will be evaluated with K-Fold Cross Validation and the model with the best result will be the optimal model. We’re looking specifically at how well the models classify new recommendations for users (accuracy) and the time measure of each model (efficiency).

**References**

Schafer, J. B., Frankowski, D., Herlocker, J., & Sen, S. (n.d.). Collaborative filtering recommender systems. *The Adaptive Web*, 291–324. https://doi.org/10.1007/978-3-540-72079-9\_9

Yehuda Koren, Robert M. Bell, Chris Volinsky: Matrix Factorization Techniques for Recommender Systems. IEEE Computer 42(8): 30-37 (2009)

Yassine, A., Mohamed, L., & Al Achhab, M. (2021). Intelligent recommender system based on unsupervised machine learning and demographic attributes. *Simulation Modelling Practice and Theory*, *107*, 102198. https://doi.org/10.1016/j.simpat.2020.102198