CS174: MACHINE LEARNING

PROJECT PROPOSAL

Personalized Song Recommender System

Michael Lau, Binjie Li, Beitong Zhang

January 22, 2015

1 PROBLEM STATEMENT

We aim to create a music recommender system based on user listening history. There are many methods for recommender systems in use today, so a major component of our project will be in determining more the appropriate methods to augment the basic Neighborhood model to fit this particular application. Our data is unique in many ways, not the least of which is the lack of explicit feedback, i.e. ratings, from the users, so we will need to survey the field to find the right way to incorporate implicit feedback. It is also clear that incorporating song features to some degree will improve the accuracy of our recommendations, so we will need to find the appropriate method to utilize these features as well.

2 METHDOLOGY

Collaborative Filtering [3,5]is one of the most successful and efficient algorithms to predict user interest and recommend highly personalized content to users in recommender systems. For our project, we have decided to use two different kinds of CF-based algorithms, *Memory-based (user-based)* and *Model-based (item-based)*.

2.1 USER-BASED COLLABORATIVE FILTERING APPROACH

The basic idea of user-based collaborative filtering approach is to recommend songs to one user u based on the set of similar users $U_s = \{u_1, u_2, .., u_k\}$. We believe that similar users would listen to similar songs and consequently we can recommend songs from the histories of the most similar users based on *neighborhood model* [4]. The similarity of the users is defined as follows:

$$SIM_{user}(i, j) = w_1 \cdot overlap(i, j) + w_2 \cdot \frac{\vec{f_i} + \vec{f_j}}{||\vec{f_i}|| \cdot ||\vec{f_j}||} \tag{2.1}$$

As shown above, parameter $overlap(i, j)$ tells the degree to which two users i, j have overlapping songs in their histories. In addition to the listening history of the user, we also consider some features about the users themselves, such as the gender, age, and nationality, which are represented by a vector $\vec{f}$.

2.2 Item-Based Collaborative Filtering Approach

We also want to recommends songs based on the similarity between their static features. If one song s has a high similarity with songs in user u's listening history, then we believe that this song is a good candidate for recommendation for the user u. We plan to use the *neighborhood model* to extract songs from the song library as the recommendation for a specific user. To achieve this goal, we need to define the similarity model for songs. Compared to users, the features and relationship of songs are more static, so we think the simple *cosine-based* similarity is appropriate for the song-based similarity computation:

$$SIM_{song}(i, j) = \frac{\vec{f_i} + \vec{f_j}}{||\vec{f_i}|| \cdot ||\vec{f_j}||} \tag{2.2}$$

As before, vector $\vec{f_i}$ represents the features of the song s_i.

2.3 Hybrid Collaborative Filtering Approach

Other than the two approaches mentioned before, we also want to try a hybrid recommendation scheme by integrating the user-based and item-based algorithms. The basic idea of this hybrid idea is we first use the user-based algorithm to choose a set of users who are similar to the target user. Among the songs in these users' histories, we use the similarity among songs to further extract a subset as the recommended songs.

3 Datasets Description

3.1 Listening history dataset from Last.fm

This dataset [2] represents the listening habits (till May, 5th 2009) for nearly 1,000 users, which was collected from Last.fm API, using the user.getRecentTracks () method. It contains two data files:

3.1.1 Listening Habits:
This dataset contains 19,150,868 lines of listening habits from 992 unique users of Last.fm. Each line has the following features:

userid;timestamp;artist-id;artist-name;track-id;track-name

3.1.2 User Profiles:
This file contains 992 profiles of users in the listening habit file. Each line represents the profile of one user, which has the following features:

userid;gender;age;country;signupdate

We would also split the dataset into two parts: training set and testing set for evaluation.

3.2 Million Songs Dataset

The Million Songs Dataset (MSD) [1] provides a variety of audio features (timbre, pitch, beats, etc.) and track information (album, artist name, location, genre, etc.) of a million contemporary popular music tracks. We will only choose a subset of this huge dataset containing more popular songs due to computational costs.

4 Milestone Goal

By the milestone, we should have built and evaluated a recommender system that uses the basic Neighborhood model described in 2.1. Hopefully we will also have begun to augment that model by incorporating song features with the method described in 2.2, and perhaps even more advanced collaborative filtering techniques.

References

[1] Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The million song dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*, 2011.

[2] O. Celma. *Music Recommendation and Discovery in the Long Tail*. Springer, 2010.

[3] Usama M Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, and Ramasamy Uthurusamy. Advances in knowledge discovery and data mining. 1996.

[4] Francesco Ricci, Lior Rokach, and Bracha Shapira. *Introduction to recommender systems handbook*. Springer, 2011.

[5] Upendra Shardanand and Pattie Maes. Social information filtering: algorithms for automating “word of mouth”. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 210–217. ACM Press/Addison-Wesley Publishing Co., 1995.