CS 475/675 Project Proposal

Xinxin Mao, Yaqian Chen, Haoxiang Li, Yawen Hu
xmao8, ychen502, hli181, yhu101

Abstract

In this project, we aimed to build a music recommendation system for users based on their listening history. We will design a hybrid approach based on both content-based filtering and collaborative filtering for the recommendation system.

1 Project choice

Choose either a **methods** or **applications** or **suggested** project, and a subarea from the below table.

☐ **Applications**

☐ Genomics data ☐ Healthcare data ☐ Text data ☐ Image data ☐ Finance data

☐ **Methods**

☐ Fairness in ML ☐ Interpretable ML ☐ Graphical Models ☐ Robust ML ☐ Privacy in ML

☐ **Suggested projects**

☒ Music AI ☐ Image Analysis ☐ Genomics & Bioinformatics

2 Introduction

A good song recommendation system can recommend songs that meet the user's preferences and enhance the user's preference and dependence on the music application. Thus, In this project, our group researches recommendation algorithms that recommend top 5 songs that are more in line with their preferences for different users depending on the hybrid of user-based method and content-based method to give a list of recommendation songs.

3 Dataset and Features

The user data is from the Taste Profile Subset released by The Echo Nest as part of the Million Song Dataset. The data comes in a (user, song, play count) triplets format and is used to build our collaborative filtering system. The training set has around one million samples with a total size of 500MB. The validation and test sets combined has 110k samples with half of their history available.
The music data comes from the Million Song Dataset, which contains a million samples with a total size of 273GB. To ease our exploration, we will use a subset of 10,000 songs (1.8GB compressed) with 12 most representative features (over 50 original features), which covers aspects including artist, genre, structure, popularity of a song, to support the build-up of our content-based filtering system.
In the pre-processing step, we will handle the json / h5 format raw data and normalize the data per user. We will further use word2vec embedding layer to densify the sparse features.

4 Methods

The algorithm has two stages: recall stage and sorting stage. 1. The purpose of the recall stage is to help us select a small number of songs from a large number of songs as candidate sets. Here, we plan to use the user-based Collaborative Filtering algorithm in this stage. User-based Collaborative Filtering algorithm predict the songs that a user might like on the basis of ratings given to that item by the other users who have similar taste with that of the target user. The user's rating of a song is calculated based on user's current song hits/her maximum song hits. 2. The purpose of the sorting phase is to sort the recalled songs, pushing the songs sequentially according to the degree of user preference. We do the Content-Based Filtering on the output of the recall stage to obtain a rating ranking. Content-based filtering compares songs' features to rank the output of recall stage, which are similar to what the user likes, based on the songs they've listend before. 3. For optimize algorithms, we can try different optimization algorithms like FTRL and SGD and compare their prediction output to find the best algorithm.

5 Deliverables

These are ordered by how important they are to the project and how thoroughly you have thought them through. You should be confident that your "must accomplish" deliverables are achievable; one or two should be completed by the time you turn in your Nov 19 progress report.

5.1 Must accomplish

1. During the recall stage, we use the user-based Collaborative Filtering to gain the top 50 similar songs.
2. In the sorting stage, do Content-Based Filtering on the output of the recall stage to obtain a rating ranking.
3. In both stages, hyperparameter automatic search algorithms (such as GridSearchCV) is used to select the optimal parameters for algorithm tuning, and the results under different optimizers (such as FTRL, SGD, etc.) are compared to obtain the best model.

5.2 Expect to accomplish

1. Add feature pre-processing, use the Word2vec embedding layer in the pre-training stage to reduce the dimension and densify the sparse features, and optimize the recommendation results.
2. In the sorting stage, look for other evaluation indicators that can be used as a good or bad ranking result, and use this indicator to obtain new recommendation results.
3. In the ranking stage, use deep learning algorithms (eg, RNN) to predict the model to improve the ranking results of the music output in the recall stage.

5.3 Would like to accomplish

1. Use Deep FM, DCN, etc. to cross the original features, find features that are more helpful to the recommendation results from more feature combinations, and further improve the recommendation results.
2. Carry out multi-objective optimization in the sorting stage, and use Multi-Task Learning (MTL) under transfer learning for training to further improve the recommendation results.
3. Further refine the training process of the recommendation algorithm, adding steps for filtering, shuffling and reordering.

References

[1] Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. *The Million Song Dataset.* In Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011), 2011.

[2] Arnold AN, Vairamuthu S (2019) *Music recommendation using collaborative filtering and deep learning.* Int J Innov Technol Explor Eng (IJITEE) 8(7):964–968

[3] Geetha G, Safa M, Fancy C, Saranya D (2018) *A hybrid approach using collaborative filtering and content based filtering for recommender system.* In: National conference on mathematical techniques and its applications (NCMTA 18), IOP publishing IOP conference series: journal of physics: conference series 1000