Personalized Music Recommendation System

Dudeja, Gautam*, Weldon, Stephen†, Maharana, Siddhartha S‡, Mahapatra, Chitta R§, Sahoo, Tanmaya K¶,
Georgia Tech

Email: *gdudeja6@gatch.edu, †sweldon6@gatech.edu, ‡smaharana3@gatech.edu, §cmahapatra6@gatech.edu
, ¶tsahoo3@gatech.edu

***Abstract*—Music recommendation engines of today uses techniques that try to mathematically measure similarities within features of recommended items such as music genre, artists, date, or measure similarities of other users to try to recommend music to a person.In this report, we present a personalized music recommendation system (PMRS) based on the convolutional neural networks (CNN) approach, that classifies music based on the audio signal beats of the music into different genres, along with collaborative filtering (CF) recommendation algorithm to combine the output of the CNN to recommend music to the user. The PMRS extracts the user's history from the user taste profile dataset and recommends music under each genre. We use the million song dataset (MSD) to evaluate our proposed solution. We used the confidence score metrics for different music genre to check the performance of our model.**

***Index Terms*—Collaborative filtering, CNN and music recommendation**

I. Introduction

Along with the rapid expansion of digital music formats, searching for songs has become significant. Two popular algorithms used in recommending music : *collaborative filtering (CF)* and *content-based filtering (CBF)*, have been found to perform well. *Collaborative filtering (CF)* models rely on usage patterns: the combinations of items that users have consumed or rated provide information about the users' preferences, and how the items relate to each other. *Content-based filtering (CBF)* approach is based on available metadata: information such as the artist, album and year of release.

II. Problem definition

Collaborative filtering, content filtering and Matrix Factorization techniques try to mathematically measure similarities within features of recommended items such as music genre, artists, date, or measure similarities of other users to try to recommend music to a person. **Limitations:** Because of the subjectivity in music, the collaborative filtering recommender does not work well due to the following key problems [14].

- Popularity bias - Generally, popular music can get more ratings. The music in long tail, however, can rarely get any. As a result, collaborative filtering mainly recommend the popular music to the listeners. Though giving popular items are reliable, it is still risky, since the user rarely get pleasantly surprised.
- Cold start - It is also known as data sparsity problems. At an early stage, few ratings is provided. Due to the lack of these ratings, prediction results are poor.
- Diversity bias - Recommendations can often be predictable, too closely related, and might not expose the user to other diverse songs that they might like.

III. Literature

An ideal music recommender system should be able to automatically recommend personalised music to human listeners [1], [2]. So far, many music discovery websites such as Last.fm, Allmusic, Pandora, Audiobaba6 , Mog7 , Musicovery 8, Spotify9, Apple "Genius" have aggregated millions of users, and the development is explosive [3], [4]. [20] proposes the easiest way to search for music using metadata information retrieval(editorial information), supplied by the creators, such as the title of the song, artist name and lyrics to find the target songs. To recommend items via the choice of other similar users, collaborative filtering technique has been proposed [5]. As one of the most successful approaches in recommendation systems, it assumes that if user X and Y rate n items similarly or have similar behaviour, they will rate or act on other items similarly. [5] further divides the collaborative filtering technique into three subcategories: memory-based [6], model-based [7] and hybrid collaborative filtering . Though collaborative fil-

tering recommenders works well, the key problems such as cold start, popularity bias are unavoidable [8].

[9] proposes a content-based(audio/signal-based) approach to make predictions by analysing the song tracks. It is rooted in information retrieval and information filtering that recommends a song which is similar to those the user has listened to in the past rather than what the user has rated 'like' [10]. Lots of research has been paid attention on extracting and comparing the acoustic features in finding perceptual similar tracks [11]. The most representative ones so far are timbre, rhythm.

[12] utilized multiple typical similarity measures like K-means clustering with Earth-Mover's Distance, Expectation-Maximization with Monte Carlo Sampling and average feature Vectors with Euclidean Distance.

IV. The dataset

The Million Song Dataset (MSD) is a freely-available collection of audio features and metadata for a million contemporary popular music tracks. The MSD is also a cluster of complementary datasets contributed by the community covering songs, lyrics, genre, song-level metadata and user data. The linked datasets that are of interest for our experiment:

- The Echo Nest Taste Profile that contains 380,000 play counts from gathered from 1 million users.
- The Last.fm Dataset that contains song tag and song similarity dataset of the MSD.

We are using MSD and user taste profile dataset to extract songs user have listened to and then use web scraper to download 30 seconds clips in mp3 format from free https://wasabi.i3s.unice.fr.

V. Proposed methodology

A. Intuition and Innovation

The key difference our recommendation engine provides that is better than the rest:

- Diverse recommendation - Can recommend similar underlying acoustics and so can be used to recommend across languages and geo-locations.
- Including both collaborative as well as content-based filtering in a single algorithm. Not just the user historical preferences but also includes deep-content based recommendations.
- Using deep neural networks(CNN) will reduce the noise signal and make more content aware recommendations like cover songs or remixed music.

B. Workflow

You can find the proposed recommendation engine workflow A of Appendix.

C. Weighted Matrix Factorization (WMF)

A dataset containing explicit rank, count or category of item or event is considered an explicit data item. 4 out of 5 rating of a movie is an explicit data point. Whereas, in implicit dataset we need to understand the interaction of users and/or events to find out its rank/category. User taste profile dataset which is part of Million song dataset (MSD) contains play counts for per user per song, which is a form of implicit dataset.

We are using WMF algorithm to learn latent factor representations of all users and items in the taste profile subset. This is a modified matrix factorization algorithm aimed at implicit feedback datasets.

Let r_{ui} be the play count for user u and song I, For each user-song pair we define a preference variable p_{ui} and a confidence variable c_{ui}

$$p_{ui} = I(r_{ui} > 0),$$
$$c_{ui} = 1 + \alpha log(1 + \epsilon^{-1} * r_{ui})$$

where α and ϵ are hyperparameters.

The preference variable indicates whether u has ever listened to song I, if its 1, we assume user enjoys the song, confidence variable measures how certain we are about that preference. It is a function of a play count.

WMF objective function is :

$$\min_{x_*, y_*} \sum_{u,i} c_{ui}(p_{ui} - x_u^T y_i)^2 + \lambda(\sum_u ||x_u||^2 + \sum_i ||y_i||^2)$$

where λ is a regularization parameter, x_u is the latent factor vector for user u, and y_i is the latent factor vector for song i. It consists of a confidence-weighted mean squared error term and an L2 regularization term. We can see first sum ranges over all users and all songs: contrary to matrix factorization for rating prediction, where terms corresponding to user-item combinations for which no rating is available can be discarded, we must take all possible combinations into account. As a result, using stochastic gradient descent for optimization is not practical for a dataset of this size. [13] propose an efficient alternating least squares (ALS) optimization method, which we are using here instead.

D. Predicting Latent Factor from music audio

1) Dataset: We are using MSD and user taste profile dataset to extract songs user have listened to and then use web scraper to download 30 seconds clips in mp3 format from free https://wasabi.i3s.unice.fr.

Prediction of latent factors for a given song from its audio signal is a regression problem. We are using librosa MIR library for feature extraction pipeline to convert music audio signals into a fixed-size representation that can be used as input to a classifier or regressor.

Extract MFCCs from the audio signals:

We are computing 13 MFCCs(Mel-frequency cepstral coefficients) from windows of 1024 audio frames, corresponding to 23 ms at a sampling rate of 22050 Hz, and a hop size of 512 samples. We also computed first and second order differences, yielding 39 coefficients in total.

2) Convolutional neural networks: With powerful representation learning abilities, Convolutional neural networks(CNN) is widely used to improve the state-of-the-art, e.g., signal processing. CNN can effectively catch local features from different layers and transform features to a single vector.

We first extracted an intermediate time-frequency representation from the audio signals to use as input to the network. We are using log-compressed mel-spectrograms with 128 components and the same window size and hop size that we used for the MFCCs (1024 and 512 audio frames respectively). The networks were trained on windows of 3 seconds sampled randomly from the audio clips. This was done primarily to speed up training. To predict the latent factors for an entire clip, we averaged over the predictions for consecutive windows.

We are working on the state-of-the-art CNN architecture, which consists of multiple layers, including rectified linear units (ReLUs), convolution, fully connected (FC), max-pooling, and 1 softmax layer.

The size of the input matrix is 1× 128 × 44, where 1 is the number of channels and each song mel spectrogram tensor is resized to 128 × 44. We remove the last FC and softmax layers, which are used for classification purposes, and take the output of the second FC layer as the representation of a song.

3) Estimating Latent Factor: Estimating latent factors for a given song from the corresponding audio signal is a regression problem. Since latent factors are real-valued, the core objective is to minimize the mean square error of the estimations. Let l_j be the latent factor vector of song j, which is obtained by WMF, and $l_j^{'}$ is the the corresponding prediction by CNN. Then, the **Objective** function is the minimization problem (θ represents the model parameters):

$$\min_{\Theta} \sum_{j} ||l_j - l_j^{'}||^2$$

This section is still under progress.

VI. Experiment and Evaluation

A. Experiment

The experiments that we have planned are trying to answer the following questions.

- Can we perform a song recommendation using the song metadata, and how does scaling the number of users, songs and metadata affect the recommendations similarity score? Does increasing the number of users, songs and metadata increase or decrease the similarity scores, and does it make some songs become less similar than others?
- Can we perform a song recommendation using the song acoustic profile, and how does scaling the number of songs affect the recommendations similarity score from one song compared to another? Does increasing the number of songs increase or decrease the similarity scores, and does it make some songs become less similar than others?
- When we combine the song metadata recommendation and song acoustic profile recommendation, are we able to increase the number of diverse recommendations? Are we able with confidence to say that we found songs that are acoustically similar to the ones recommended but are dissimilar from the song metadata similarity score.

This section is still under progress.

B. Evaluation

When evaluating the music recommendation methods there are several metrics that we would like to explore and try to optimize.

- Finding the most impactful features to use when performing song metadata and song acoustic recommendations. We intend to continue to add features as we scale the infrastructure up to handle more songs and users and finding the most important features to use (such as genre, artist location, language), would improve the recommendation similarity score and reduce the number of unnecessary features, which reduces computational complexity and overfitting.

- Finding optimal hyperparameters for use in the Matrix Factorizations and Convolutional Neural Networks. By optimizing these hyperparameters we will be able to create more impactful recommendations.
- Perform a statistical analysis on song similarity compared from the song metadata and acoustic profile and see the ratios of how many songs are extremely different from each other or very similar to each other.
- Increase the number of diverse song recommendations with higher diversity ratings. We want to find as many as songs as possible to recommend that are very dissimilar from a song metadata perspective, but very similar acoustically. To do this, we will need to scale the number of songs we're considering and be able to accurately measure song similarity and acoustic similarity.
- Test the recommendation portion using hold out sets on the user's profiles. By randomizing the removal of songs from their taste profiles and then recommending other songs based off songs in their profile we want to see if the recommendation we generate for that user's songs are accurate or not. We want to guess if the randomized song that was removed is inside the recommendations we present.

This section is still under progress.

VII. Conclusions and Discussion

This section is still under progress.

VIII. Milestone, Responsibilities and Status

Table I summarizes the deliverable and responsibilities for each team member along with status.

REFERENCES

[1] Paul Lamere. 2008. “Social tagging and music information retrieval”. Journal of new music research37, 2 (2008), 101–114.

[2] François Pachet, Daniel Cazaly, et al. 2000. “A taxonomy of musical genres”. In RIAO. Citeseer, 1238–1245.

[3] Pedro Cano, Markus Koppenberger, and Nicolas Wack. 2005. “An industrial-strength content-based music recommendation system”. In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. 673–673.

[4] Pedro Cano, Markus Koppenberger, and Nicolas Wack. 2005. “Content-based music audio recommendation”. In Proceedings of the 13th annual ACM international conference on Multimedia.211–212.

[5] Diego Sánchez-Moreno, Ana B Gil González, M Do-lores Muñoz Vicente, Vivian F López Batista and María N Moreno García. 2016. “A collaborative filtering method for music recommendation using playing coefficients for artist sand users”. Expert Systems with Applications 66 (2016), 234–244.

[6] John S Breese, David Heckerman, and Carl Kadie. 2013. “Empirical analysis of predictive algorithms for collaborative filtering”.arXiv preprint arXiv:1301.7363(2013).

[7] Mustansar Ali Ghazanfar, Adam Prügel-Bennett, and Sandor Szedmak. 2012. “Kernel-mapping recommender system algorithms”.Information Sciences 208 (2012), 81–104 .

[8] Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen and John T Riedl. 2004. “Evaluating collaborative filtering recommender systems”. ACM Transactions on Information Systems (TOIS)22, 1 (2004), 5–53.

[9] Qing Li, Byeong Man Kim, Dong Hai Guan, and Duk whan Oh. 2004. “A music recommender based on audio features”.In Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval.532–533.

[10] Beth Logan. 2004. “Music Recommendation from Song Sets”.In ISMIR. 425–428.

[11] Chun-Man Mak, Tan Lee, Suman Senapati, Yu Ting Yeung and Wang-Kong Lam. 2010. “Similarity Measures for Chinese Pop Music Based on Low-level Audio Signal Attributes”. In ISMIR. 513–518.

[12] Terence Magno and Carl Sable. 2008. “A Comparison of Signal Based Music Recommendation to Genre Labels, Collaborative Filtering, Musicological Analysis, Human Recommendation and Random Baseline”. In ISMIR. 161–166.

[13] Yifan Hu, Yehuda Koren, and Chris Volinsky. “Collaborative filtering for implicit feedback datasets”. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, 2008.

[14] Jonathan L. Herlocker, Joseph a. Konstan, Loren G. Terveen, and John T. Riedl. “Evaluating Collaborative Filtering Recommender Systems”. ACM Transactions on Information Systems, 22(1):5–53, January 2004

[15] Ke Chen, Beici Liang, Xiaoshuan Ma and Minwei Gu. “Learning Audio Embeddings with User Listening Data For CONTENT-BASED MUSIC RECOMMENDATION”.

[16] Thaair Ameen, Ling Chen, Zhenxing Xu, Dandan Lyu and Hongyu Shi. “A Convolutional Neural Network and Matrix Factorization-Based Travel Location Recommendation Method Using Community-Contributed Geotagged Photos”. ISPRS Int. J. Geo-Inf. July 2020, 9(8), 464.

[17] Dawen Liang, Minshu Zhan, and Daniel P. W. Ellis. “Content-Aware Collaborative Music Recommendation Using Pre-trained Neural Networks”. Published in ISMIR 2015.

APPENDIX

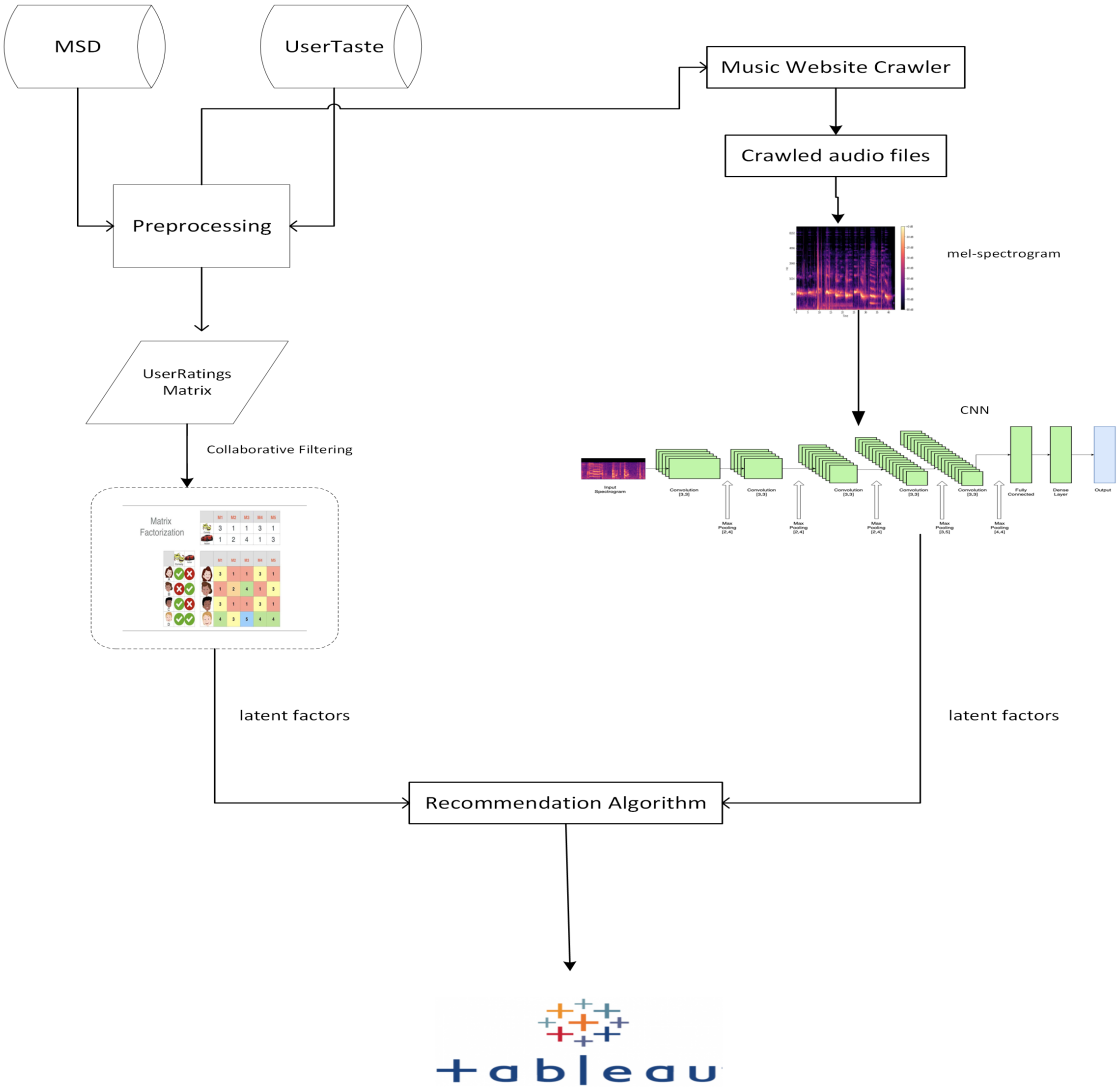

Fig. 1. proposed recommendation engine workflow

TABLE I
MILESTONE, RESPONSIBILITIES AND STATUS

Deliverables	Responsibility	Initial Timeline	Revised Timeline	Status
Exploratory Data Analysis	Gautam, Stephen	10/15/2021	10/15/2021	100%
Presentation, Video	Gautam, Stephen	10/15/2021	10/15/2021	100%
Project Proposal	Chitta(lead), Everybody	10/15/2021	10/15/2021	100%
Project Progress report	Chitta(lead), Everybody	11/05/2021	11/07/2021	100%
Infrastructure Setup	Chitta, Siddhartha, Tanmaya	11/05/2021	11/08/2021	10%
Data Extraction	Everyone	11/05/2021	11/15/2021	10%
Data pre-processing, feature extraction(data profiling)	Stephen(lead), everybody	11/05/2021	11/15/2021	50%
Model Building	Gautam, Siddhartha, Stephen	11/15/2021	11/15/2021	50%
Interactive User Interface	Siddhartha, Tanmaya, Chitta	11/15/2021	11/20/2021	0%
Project Report	Chitta(lead), everybody	11/25/2021	12/03/2021	25%
Final Poster	Everybody	12/03/2021	12/03/2021	0%