

# 643 Final Project Proposal: A Multi-faceted Approach to Music Recommendations

*Authors: Robert Sellers, Logan Thompson, James Topor*

*July 7, 2017*

## Introduction

Web-based music streaming services such as **Spotify**, **Pandora**, **SoundCloud**, and **Tidal** provide their users with many opportunities to discover new music, whether in the form of specific pieces of music the user hadn't heard before or in the form of musical artists to which the user hasn't previously been exposed. These systems make use of tools such as collaborative filtering and content-based filtering as part of their efforts to further engage their user base. Given the widespread use of such methodologies for enabling the discovery of new music, today's web-based streaming environment offers ample opportunity for those interested in exploring both the methods typically used for constructing recommender systems and how such systems can effectively be applied to enable the discovery of novel content.

---

## Problem Formulation & Objectives

The purpose of this project will be to implement a multi-faceted approach to musical artist recommendations through the use of a user-based collaborative filtering algorithm, similarity matrices, content-based filtering, and an interactive application interface. The goal of the project will be to gain experience in the implementation of a variety of recommendation algorithms using a large (1M+ item) data set and to gain insight into how many commercial recommender systems enable "user discovery" of different content. Additionally, this project will provide the authors with hands-on experience in implementing an interactive user interface within a combined collaborative/content-based recommender system framework.

---

## Approach

The project will be implemented using **R** / **RStudio**, **Shiny**, **Github**, and the **last.fm** publicly available dataset of system user, musical artist, and user-supplied music genre labelings. A "Top N" user-based collaborative filter, artist-genre matrix, and artist similarity matrix will each be constructed within **R**. The collaborative filter will be constructed using the **recommenderlab** toolset and will generate a "Top N" list of recommended artists for each **last.fm** user. The resulting data structure will be saved within an **RData** file and uploaded to **Github** for use within an envisioned Shiny application. Similarly, the artist similarity and artist-genre matrices will also be saved as **RData** files and uploaded to **Github** for use within an envisioned **Shiny** application.

An envisioned **Shiny** application will allow a prospective **last.fm** user to do each of the following:

1. Review the list of artists to which they've previously listened;
2. View the "Top 10" artist recommendations generated by the user-based collaborative filtering recommender system;

3. Receive a “Top 5” list of suggested of musical artists who are likely similar to an artist specified/selected by the user;
4. Receive a “Top 5” list of suggested of artists for a user-selected musical genre.
5. For any artist listed in the results of items 1, 2, 3, or 4 (above), the user will be able to simply select the artist’s name and click an icon to activate item 3, thereby generating a new “Top N” list of suggested musical artists who are likely similar to the selected artist.

A mockup of the envisioned **Shiny** interface is shown below.

### last.fm ARTIST RECOMMENDER

ENTER USER ID		RESULTS
1234		
CHOOSE FILTER		
<input type="radio"/>	REVIEW PREVIOUS LISTENS BY ARTIST	
ARTIST RECOMMENDATIONS		
<input checked="" type="radio"/>	BY SIMILAR USERS [10 TOTAL]	
<input type="radio"/>	BY SIMILAR ARTISTS [TOP 5]	
<input type="radio"/>	BY GENRE [TOP 5]	
UPDATE VIEW		ARTIST NAME 1
		ARTIST NAME 2
		ARTIST NAME 3
		ARTIST NAME 4
		ARTIST NAME 5
		SELECT AN ARTIST NAME FOR MORE RECOMMENDATIONS

Figure 1:

## Data to be Used

The data set to be used is comprised of music listening information for a set of 1,892 users of the **Last.fm** online music system. The data set is contained within a series of text files, the following of which will be used as part of this project:

**tags.dat:** This file contains a list of musical genres that **last.fm** users have used to categorize the various musical artists represented within the **last.fm** online music streaming platform. Each genre is assigned a unique identifier, or “tagID”. The following sample of data from the **tags.dat** file shows that each musical genre and unique tagID pair is provided in a separate row, with the data values delineated by spaces:

```
tagID tagValue
1 metal
```

2 alternative metal  
3 goth rock  
4 black metal  
5 death metal

**artists.dat**: This file contains a list of musical artists available within the **last.fm** platform. Each artist is assigned a unique identifier, or “artistID”. Furthermore, a URL to a webpage for the artist as well as a second URL to a photo of the artist are provided. For purposes of this project, we will not make use of either of the two URLs.

The structure of the file is as follows:

id name url pictureURL

As with the **tags.dat** file, items within the file are separated by spaces.

**user\_artists.dat**: This file lists the artists to which each user has listened and also provides a “listen count” for each [user, artist] pair. A total of 17,632 distinct musical artists are represented within the data set, resulting in a total of 92,834 [user-listened artist] pairs.

The file sample below shows that each user is assigned a unique ID, and the artists to which they have listened are represented by their respective artistID’s. The number of times a user has listened to an artist is represented by the value contained within the “weight” column.

userID artistID weight  
2 51 13883  
2 52 11690  
2 53 11351  
2 54 10300  
2 55 8983

**user\_taggedartists.dat**: This file contains a listing of each instance in which a **last.fm** user has assigned a musical genre label (a.k.a., a “tag”) to an artist. The file also contains the date (day, month, and year) of the “tagging”. As can be seen in the small file sample below, each user can apply more than one genre label to any given artist.

userID artistID tagID day month year  
2 52 13 1 4 2009  
2 52 15 1 4 2009  
2 52 18 1 4 2009  
2 52 21 1 4 2009  
2 52 41 1 4 2009  
2 63 13 1 4 2009

The date components of the file will not be made use of for this project.

---

## Combining the Data

We envision combining the data contained within these various files in different ways to enable our proposed application:

1. Content from **artists.dat** and **user\_artists.dat** will serve as the basis of the proposed “Top N” collaborative filter;
2. Content from **tags.dat**, **user\_taggedartists.dat**, and **artists.dat** will serve as the basis of a artist-genre matrix;

3. Content from **tags.dat**, **user\_taggedartists.dat**, and **artists.dat** will serve as the basis of an “artist similarity” matrix and related “Top N” similar artists list.

By combining the data in such ways, we hope to be able to provide the users of our envisioned **Shiny** application with a variety of ways in which to discover new music.