Yahoo Music Recommendation

Group – CodeChef

Project Report

Under the Guidance of

Prof. Rensheng Wang

By

Amit Bhorania
Chiranth HD
Vandana Velagala

Stevens Institute of Technology

Dept. of Electrical & Computer Engineering

Data Acquisition & Processing I (EE-627A)

CERTIFICATE

This is to certify that the project report entitled “Yahoo Music Recommendation” done by Amit Bhorania, Chiranth HD and Vandana Velagala is an authentic work carried out by them at Stevens Institute of Technology, Hoboken, NJ, USA under my guidance. The matter embodied in this project work has not been submitted earlier for the award of any other degree to the best of my knowledge and belief.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without mentioning of the people whose constant guidance and encouragement made it possible. We take pleasure in presenting before you, our project, which is result of studied blend of both research and knowledge. We express our earnest gratitude to our project guide, **Professor Rensheng Wang**, Department of Electrical Engineering, for his constant support, encouragement and guidance. We are grateful for his cooperation and his valuable suggestions. We would also like to thank you our **TA Mr. Meng Cao** for providing the sample code for starting this project and helping us throughout the semester whenever help needed.

TABLE OF CONTENTS

LIST OF FIGURES

1 Introduction

1.1 Project Definition

The goal of the project is to design and implement Music Recommendation system that works on user data and ratings and provides recommendation for songs. It then processes the input data using various algorithms. Then the algorithm predicts the recommendations for songs. This project uses Yahoo Music Data set which is very sparse. The project is implemented using the Python Language.

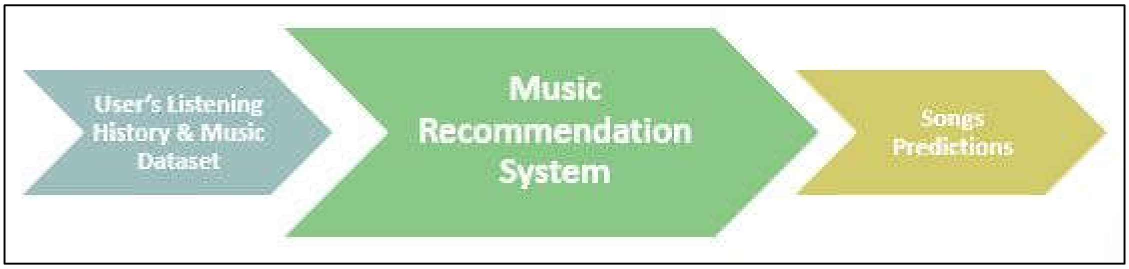

User's Listening History & Music Dataset
Music Recommendation System
Songs Predictions

Figure 1 Project Idea

1.2 Motivation

Nowadays, music plays a very crucial part in our life. Some people literally eat, live and breathe music. But as we know the music data around us is very large. The number of songs available exceeds the listening capacity of single individual. People sometimes feel difficult to choose from millions of songs. Moreover, music service providers need an efficient way to manage songs and help their costumers to discover music by giving quality recommendation. People listen to a variety of songs on various website like Pandora, EMusic, Spotify, etc., and all of them have their recommendation system or in other way we can say they recommend songs to the user based on some parameters.

Well, that is an interesting thing. We tried to find out how only a few songs are selected and recommended from such a large data. So for this project, we designed a music recommendation system.

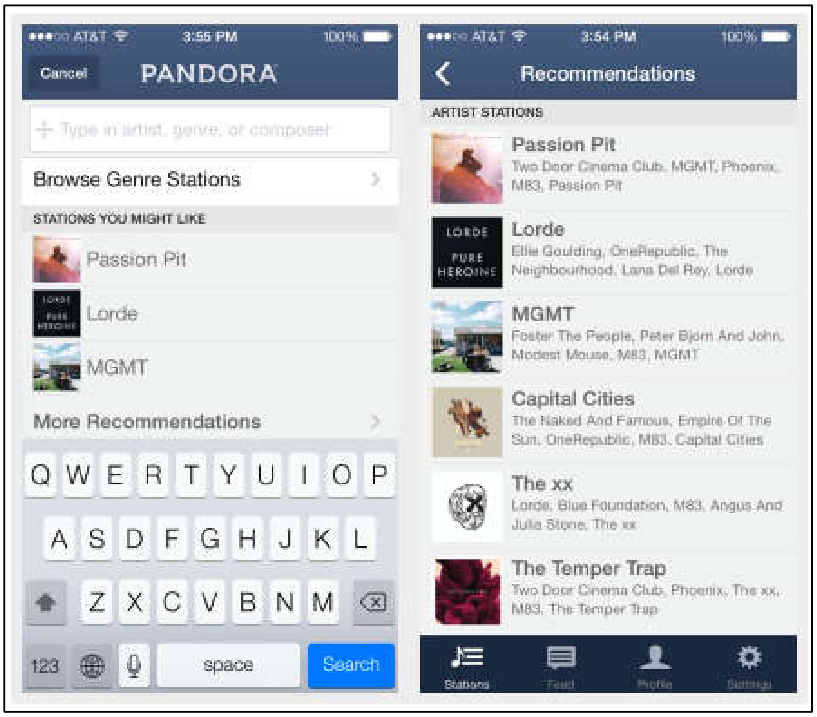

AT&T 3:55 PM 100%
Cancel
PANDORA
Type in artist, genre, or composer
Browse Genre Stations
STATIONS YOU MIGHT LIKE
Passion Pit
Lorde
MGMT
More Recommendations
Q W E R T Y U I O P
A S D F G H J K L
Z X C V B N M
123
space
Search
AT&T 3:54 PM 100%
Recommendations
ARTIST STATIONS
Passion Pit
Two Door Cinema Club, MGMT, Phoenix, M83, Passion Pit
Lorde
Ellie Goulding, OneRepublic, The Neighbourhood, Lana Del Rey, Lorde
MGMT
Foster The People, Peter Bjorn And John, Modest Mouse, M83, MGMT
Capital Cities
The Naked And Famous, Empire Of The Sun, OneRepublic, M83, Capital Cities
The xx
Lorde, Blue Foundation, M83, Angus And Julia Stone, The xx
The Temper Trap
Two Door Cinema Club, Phoenix, The xx, M83, The Temper Trap
Stations
Feed
Profile
Settings

Figure 2 Music Recommendation Example

1.3 Scope of Work

Tasks that we had in hand are:

- Understanding Yahoo Music Data Set
  - Yahoo music data set is very large and contains many parameters like the Users, Tracks, Album, Artist, Genre and Ratings. Every parameter in the dataset is denoted by a specific ID.
- Data Processing
  - The data which we get online is usually not fit to use. Yahoo Music data was unsorted so we cannot use it directly. In order to solve this problem, we pre-process the data so that it can be used.
- Training Algorithm
  - We pass the data which we get after data processing through the training algorithm to train the algorithm and get the training output.
- Testing Dataset
  - Our testing dataset contains six tracks for each user which are to be tested.
- Observation
  - After the data is tested observations are made and output is predicted.

2 Data Set

Yahoo Music Dataset is very sparse and contains many parameters like user id, track data, album, artist, genre and rating information.

Dataset consists of approximately

- 40000 users
- 224000 songs
- 53000 albums
- 18700 artist
- 600 genre

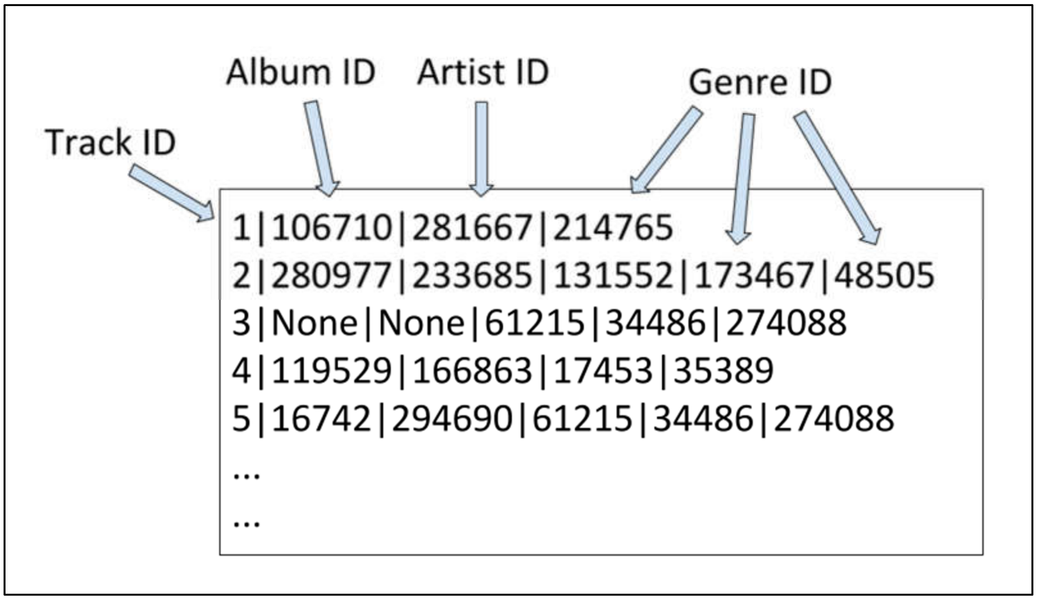
The Track data consists of Hierarchy of

Track => Album => Artist => Genre

Important Dataset information is provided below with pictorial view.

2.1 Track Dataset

Track Dataset contains id of a particular track and its corresponding Album ID, Artist ID and Genres ID. We could have multiple genre id as a track can be associated with many genres.

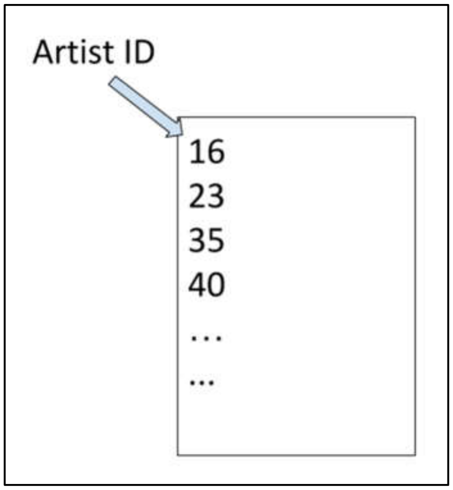


```
1|106710|281667|214765
2|280977|233685|131552|173467|48505
3|None|None|61215|34486|274088
4|119529|166863|17453|35389
5|16742|294690|61215|34486|274088
...
...
```

Figure 3 Track Hierarchy Information

2.2 Album Artist & Genre IDs

Data set contains separate files for album, artist and genres which represents the IDs of album, artist and genres present in dataset. This was useful while deriving a hierarchy algorithm where a track was suggested using the corresponding album, artists and genres. It also helps to check whether the item id in the training data is of which type.

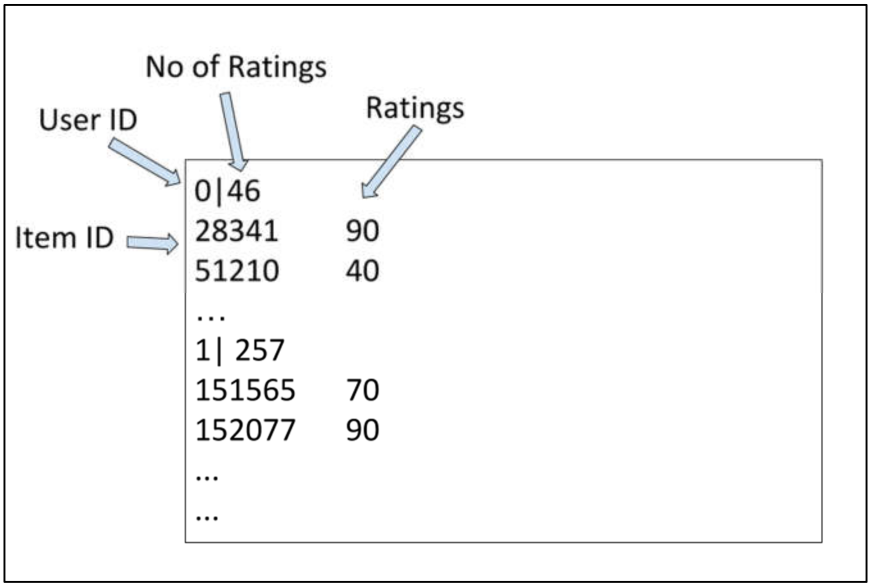


```
16
23
35
40
…
…
```

Figure 4 Album Artist & Genre IDs

2.3 Training Data

Training data contains the user id and number of items the user had rated. The rating could be for a track, album, artist or genre. This Training data will be used to Train the Algorithm and predict the songs for the user.

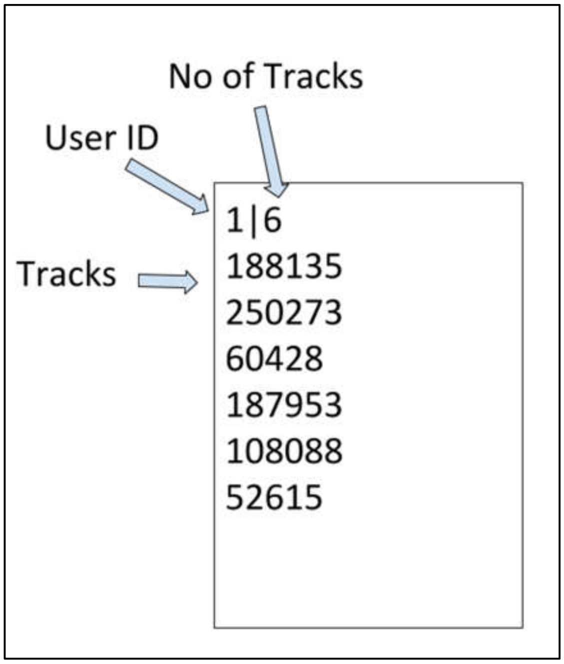


```
0|46
28341       90
51210       40
…
1| 257
151565      70
152077      90
…
…
```

Figure 5 Training Data

2.4 Testing Data

Based on the training, algorithm was provided below test data file which includes the IDs of a song to predict. Algorithm will predicts the ratings for the 6 tracks and sort them according to ratings value. The 3 highest rating tracks will be recommended for user and will be given 1 as result. Other 3 tracks will be given 0 value as result.



```
1|6
188135
250273
60428
187953
108088
52615
```

Figure 6 Testing Data

2.5 Data Pre-Processing

Raw data which we get is not always useful for training. We need to process the given data and generate new data for Machine Learning.

We extracted below data for use in Algorithms

- Track => Album => Artist => Genre Hierarchy for each Testing Data
- List of Tracks presents in Each Album (Album => Tracks)
- List of Albums of each Artist (Artist => Album)
- List of Artist of various Genres (Genre => Artist)

3 Training Algorithm

The crucial part of any Machine Learning Project is to select the Algorithm for Data Processing & Training. We started with basic approach to use only Album and Artist information and we implemented one after one approach from weights to Ensemble Algorithms to use various algorithms to solve the given problem.

This section provides insights of the various algorithms used in the project.

3.1 Method 1 – Album & Artist Ratings

- This algorithm considers both the album and artist data file.
- It takes all the ratings of the particular album and corresponding artist file.
- The Approach is to add the Ratings of Album and Artist.
- If any value is not present then it will be replace by None Value
- Initially we use the None value as 50 and later we varied from 0 to 100 in step of 10 to check the effect on the result
- Below is the result for all the values
    - None Value = 0 – 0.8440
    - None Value = 10 – 0.8413
    - None Value = 20 – 0.8400
    - None Value = 30 – 0.8335
    - None Value = 40 – 0.8278
    - None Value = 50 – 0.8046
    - None Value = 60 – 0.7848
    - None Value = 70 – 0.7425
    - None Value = 80 – 0.6914
    - None Value = 90 – 0.4710
    - None Value = 100 – 0.2146
- We can observe that None value as 0 gives the highest prediction 0.8440 from all this approaches

3.2 Method 2 – Album + Artist + Genre Ratings

- In this method, we considered genre along with the album and artist ratings.
- Same procedure was followed as method 1.
- Obtained an accuracy of 0.7953

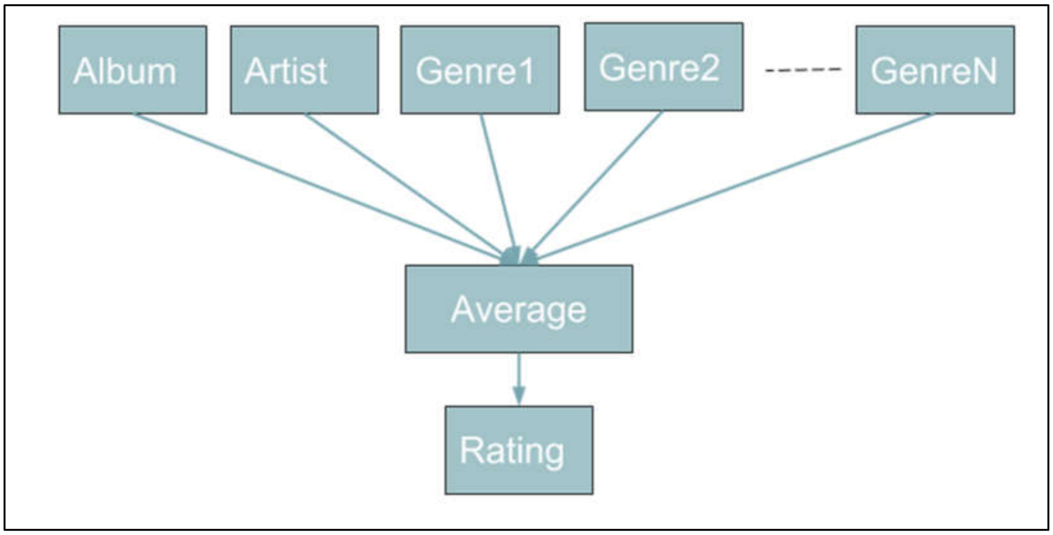


Figure 7 Album Artist and Genre Average Algorithm

3.3 Method 3 – Album + Artist + Genre Ratings with weights

- Album is more related to Songs compared to Genre
- Album ratings will have more impact on recommendation
- So we use the weights for each information
- The weights were changed to get the best possible result.
- So here changing the weights gives the change in correction rate
- Album (0.4), Artist (0.3), Genre (0.3) – 0.8645
- Album (0.5), Artist (0.3), Genre (0.2) – 0.8655
- Album (0.7), Artist (0.2), Genre (0.1) – 0.8588

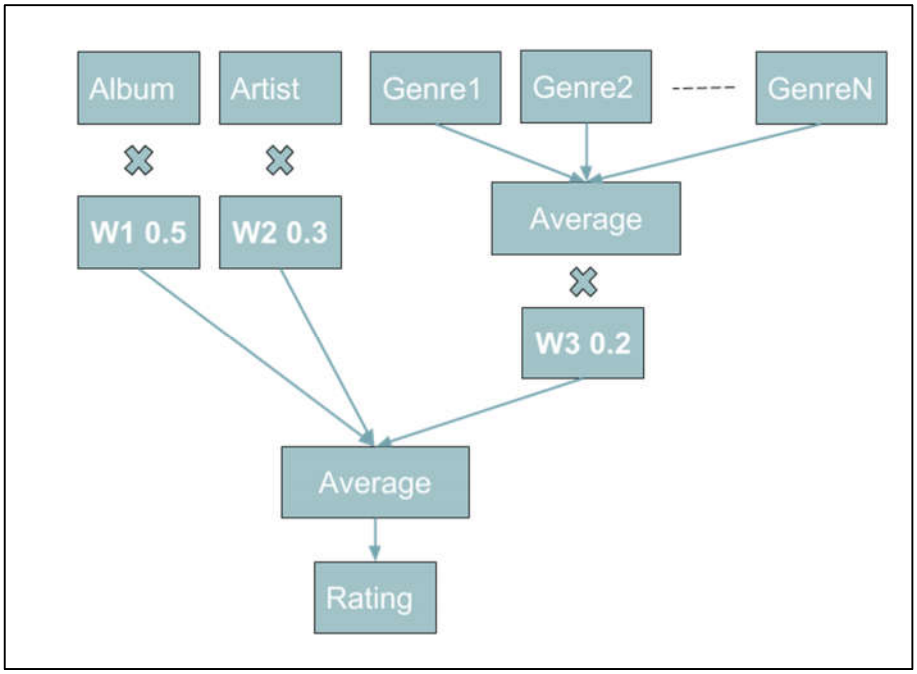


Figure 8 Album Artist and Genre with Weight Algorithm

3.4 Method 4 – Forward & Backward Hierarchy Algorithm

- Track has hierarchy with Album, Artist and Genre
- First we check whether the track is contained in any Album or not, then we check the album’s rating and also the ratings of other songs contained in the album.
- Then we check whether the album is sung by any Artist, then we check the rating of the Artist and also the rating of any other album sung by the particular Artist.
- Then we check whether the Song is contained in any genre by the Artist, if yes we check its rating and the rating of all other genres by the and also the other artists in those genres.
- Then we process the information obtained and get an output.
- Accuracy – 0.8566

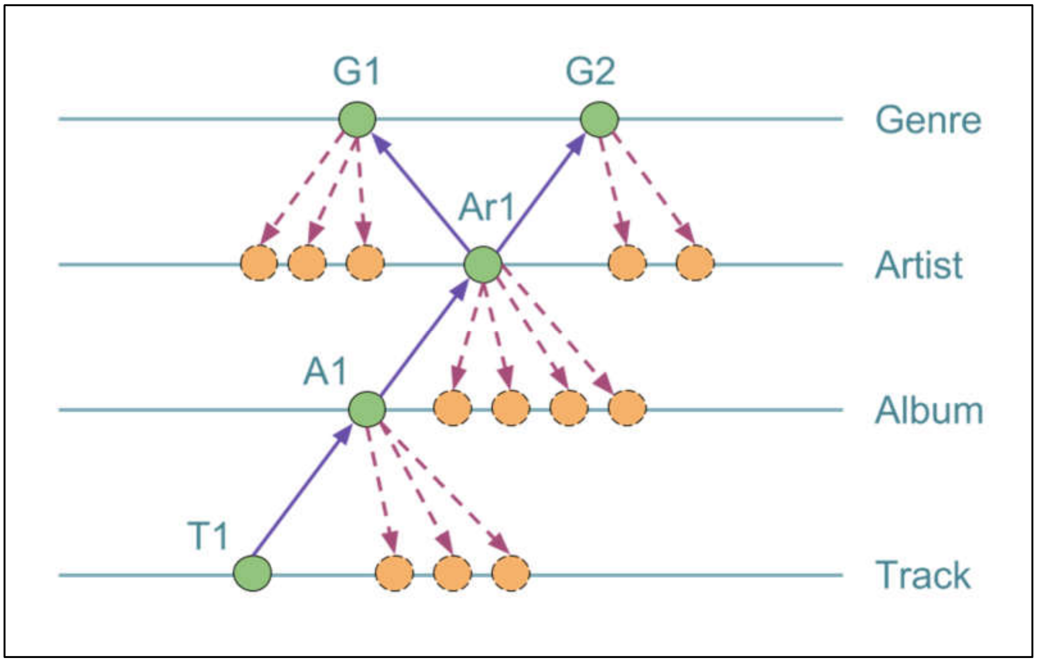


Figure 9 Forward & Backward Algorithm

3.5 Method 5 – Forward & Backward Hierarchy Algorithm with weights

- Here, the forward & backward algorithm is used with weights for each level
- Album Level is given more weight as it impacts more on the Track Data
- Artist Level is given some lesser weight as it impacts moderate on the Track Data
- At last, Genre is given small weights as its impact is very little on song prediction
- Below are the observation for various weights
- Album Level (0.7), Artist Level (0.25), Genre Level (0.05) – 0.8678
- Album Level (0.6), Artist Level (0.3), Genre Level (0.1) – 0.8676
- Album Level (0.8), Artist Level (0.2), Genre Level (0.0) – 0.8460

3.6 Method 6 – Ensemble Algorithm

- Once we have observation from various algorithms, we can ensemble the results of various algorithm and generate the new prediction.
- This method is helpful because various algorithms have their own approaches and Pros and cons. Therefore sometimes a method can have corner cases which it cannot predict properly.
- In this scenario, if we ensemble different outputs then one method can eliminate the corner cases of other methods and can help each other to improve the result.
- Here, we tried the prediction results from the various previous results and tried combine two to up to ten prediction files results using Ensemble Algorithm
- We got prediction results from 0.7 to 0.87 for different cases.
- The Highest prediction result that we got is 0.8798

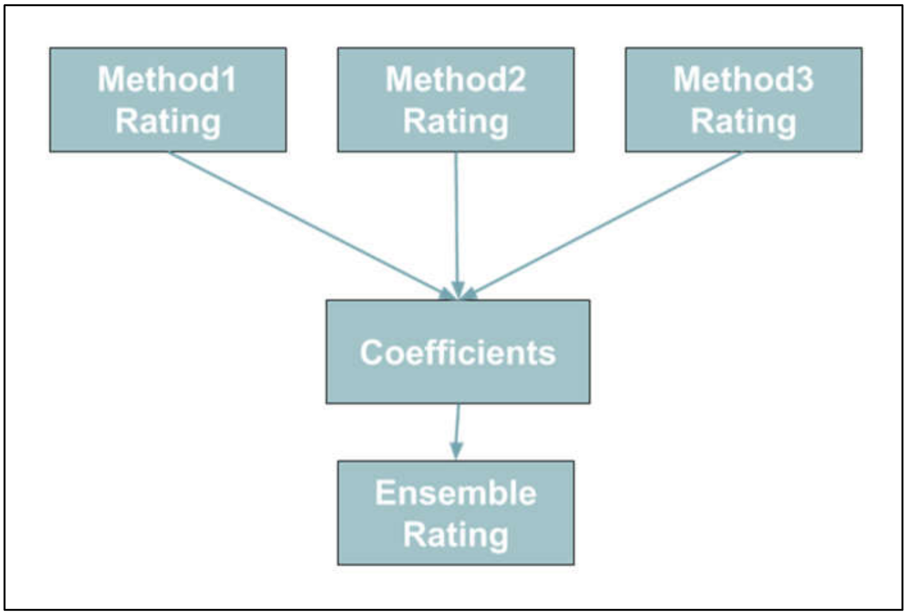


Figure 10 Ensemble Algorithm

4 Result

Below Table provides the information of prediction results and its correction rate

No	Decision File Name	Method Used	Correction Rate
1	M1_pred_Alb_Art_0	Method 1 (None Value = 0)	0.8440
2	M1_pred_Alb_Art_50	Method 1 (None Value = 50)	0.8046
3	M1_pred_Alb_Art_100	Method 1 (None Value = 100)	0.2146
4	M2_pred_Alb_Art_Genre	Method 2	0.7953
5	M3_pred_weight_4_3_3	Method 3 (weight – 0.4, 0.3, 0.3)	0.8645
6	M3_pred_weight_5_3_2	Method 3 (weight – 0.5, 0.3, 0.2)	0.8655
7	M3_pred_weight_7_2_1	Method 3 (weight – 0.7, 0.2, 0.1)	0.8588
8	M4_pred_Forward_Backward	Method 4	0.8566
9	M5_pred_6_3_1	Method 5 (weight – 0.6, 0.3, 0.1)	0.8676
10	M5_pred_7_025_05	Method 5 (weight – 0.7, 0.25, 0.05)	0.8678
11	M5_pred_8_2_0	Method 5 (weight – 0.8, 0.2, 0.0)	0.8460
12	M6_Ensemble1	Method6	0.8742
13	M6_Ensemble2	Method6	0.8798

Observation

- The result table shows that the highest correction rate is 0.8798 by using Method 6
- Here for Method 3 and Method 5, different weights are used. We can also observe that the correction rate also changes according to weight used
- Compare the Method 2 and 3. Both uses the sum of all album, artist and genre ratings but one uses the normal average and second uses weighted average. We can observe that with weights, result improves much.
- Ensemble Algorithm used on different approaches provides better result than the individual ones. It can be seen from Prediction files of 12 and 13.

5 Conclusion

Working on this project was a great learning experience as well as helped us to get an opportunity to work on the practical example of Machine Learning Application. We made our hands dirty with Data Pre-processing, find different approaches to improve the correction rate and wrote completely new algorithm from scratch. Leader Board also helped us to have healthy competition among our class groups which motivated us to achieve higher scores.