Music Recommendation System Using Content based Analysis of Song meta data

Himaja Rachakonda
New York University
New York, USA
himaja.rachakonda@nyu.edu

Manasa Kunaparaju
New York University
New York, USA
mk5376@nyu.edu

Niharika Kunaparaju
New York University
New York, USA
niharika.kunaparaju@nyu.edu

I. ABSTRACT -

In our project, we decided to experiment, design and implement a song recommendation system. We aim to do this by formulating a similarity distance metric between songs based on three semantic descriptive vectors and validate this against the User's Taste Profile Dataset. We aim to achieve this by performing content based analysis of audio spectral properties. We used the million song dataset which has an extensive collection of audio meta data. We wanted to experiment this approach to eliminate the cold start problem faced in collaborative filtering for music recommendations.

Keywords – Analytics, MapReduce, Hadoop, Hive, Spark, Recommender, Clustering

II. INTRODUCTION

In recent years, the music industry has shifted more and more towards digital distribution through online music stores and streaming services. As a result, automatic music recommendation has become an increasingly relevant problem: it allows listeners to discover new music that matches their tastes, and enables online music stores to target their wares to the right audience.

Although recommender systems have been studied extensively, the problem of music recommendation in particular is complicated by the sheer variety of different styles and genres, as well as social and geographic factors that influence listener preferences. The number of different items that can be recommended is very large, especially when recommending individual songs. Broadly, there are two main approaches to recommend items to users: collaborative filtering and content-based filtering.

The goal of this project is to build a recommender system by formulating a similarity distance metric between songs based on three semantic descriptive vectors – "Loudness", "timbre", and "pitch". We also would want to validate this against the User's Taste Profile Dataset.

III. MOTIVATION

With the advent of online services such as Spotify and Pandora, the music industry has shifted away from the traditional distribution model of selling physical copies of music to a cloud-based model that provides music for users to listen to. In this model, value is derived when these services present songs that the customer is interested in; either the customer purchases a subscription, or the customer pays for the song. In both cases, these music services can derive financial gain by improving their recommendations to potential customers of the songs they like. Thus, there is strong financial incentive to implement a good song recommendation system.

We have observed that Collaborative Filtering approach does not provide accurate results when there is no considerable amount of user data available. This becomes a problem for a lot of new artists as their music will not be recommended as much as a popular artist's music would have been recommended. Also when a new user signs up into web sites like Pandora or Spotify, collaborative filtering will not provide accurate results as the user will not have an existing saved playlist.

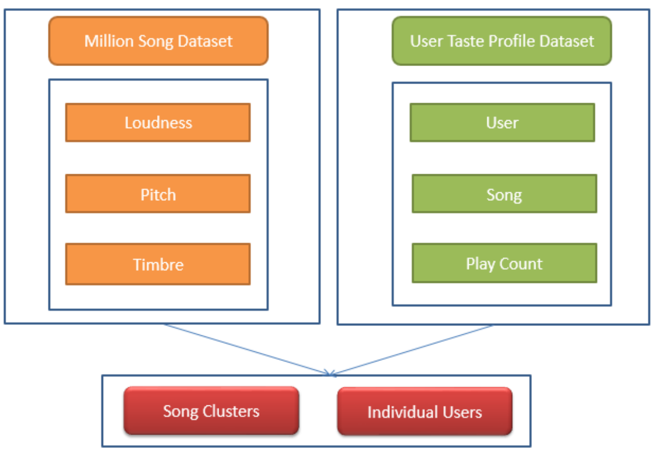
Content based analysis helps in recommending songs based on user's query for a song. We wanted to analyse, if we can recommend an accurate suggestion based on the low level audio features of the queried song. We wanted to experiment if we can eliminate the cold start problem in this way. As the songs on the playlist increase for a particular user, recommendation can be weighed against the low level features of all the songs in the playlist, along with the play count and skip count data, in the long run.

IV. RELATED WORK

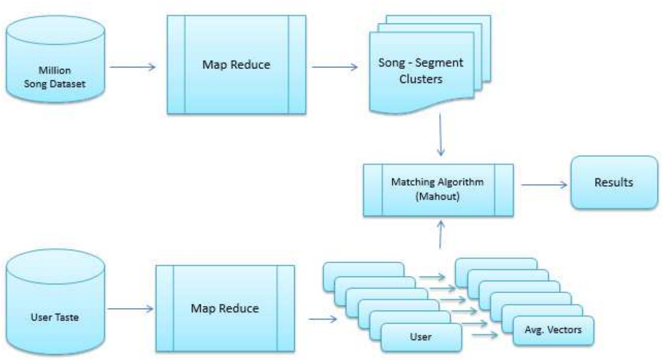
In preparation for this project we researched on a few papers which provide us with an insight on how to go about solving the problem of music recommendation system. The paper Million Song Data Set Recommendation [8] tells us about a novel way to recommend songs that a user might enjoy. They have used KNN and matrix factorization. From this we have concluded that KNN would be effective in clustering the user dataset. The paper learning content similarity for Music Recommendation [6] proposed a method for optimizing content-based similarity by learning from a sample of collaborative filter data. The optimized content-based similarity metric can then be applied to answer queries on novel and unpopular items, while still maintaining high recommendation accuracy. The paper song genre classification [7] revolves around automatic song classification by genre. The paper proved that models like Hidden Markov Model are not particularly effective in creating a recommendation system. Various other papers are particularly useful in the implementation aspect of the project. The scalable hierarchical clustering algorithm [5] paper helps us find out a way to implement the user and song attribute clustering.

V. Design

Music Recommendation System Using Content based Analysis of Song Similarity - Data



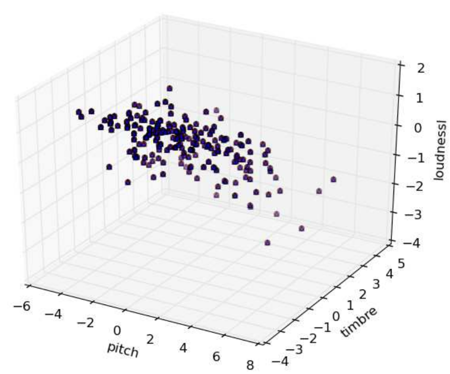
Music Recommendation System Using Content based Analysis of Song Similarity - Architecture



We primarily used two datasets provided by echonest – the Million Song Data Set and the User Taste Profile dataset. The Million Song Dataset is a freely-available collection of audio features and metadata for a million contemporary popular music tracks. The core of the dataset is the feature analysis and metadata for one million songs, provided by The Echo Nest. Taste Profile dataset contains real user - play counts from undisclosed partners, all songs matched to the Million Song Dataset. It has a 48,373,586 user-song –play count triplets.

Audio files can be parameterized into mel-scaled cepstral coefficients. Similarity measure is a fundamental step in content based audio analysis such as audio classification, audio retrieval and audio scene analysis. In most of current audio analysis systems, similarity measure is based on statistical characteristics of the temporal and spectral features of each frame, and the statistics including mean, and standard deviation or covariance are used to describe the property of an audio clip. These statistical features have proved their effectivity in many previous works[9]

As it is proved from previous research that the spectral features of a song provide a good similarity metric, we have chosen the segment features of a song as primary features of our analytic viz. Loudenss, Pitch and Timbre (MFCC like). The following is the 3-D plot of the spectral features of the songs in the Million song dataset.

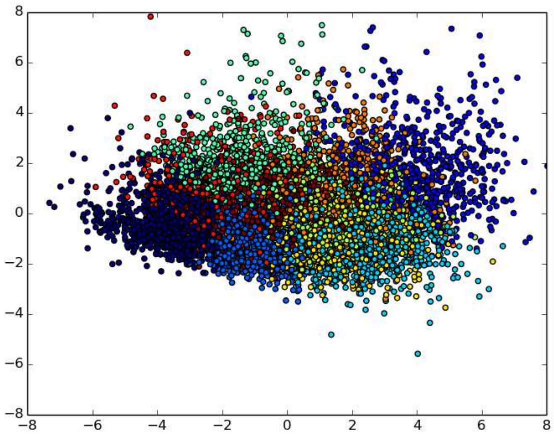


The User taste profile data set has the triplets of userIds, songIds and play counts. This data is used to suggest recommendations to a user based on the limited playcount history. As User taste profile data set's songIds are matched to the Million song dataset; we designed our algorithm to compute distances of user's song listings to the songs in the million song dataset. For this, we processed each of these datasets seperately and applied the user's information on the million song dataset via a matching algorithm.

VI. Results

We performed clustering on all the songs based on the loudness, pitch and timbre attributes using K-means Clustering. We classified the songs into ten different clusters to recommend new songs based on a queried song/limited playlist. The User taste profile data set is proceesed to obtain the user's listening history. Each user's listeing history is then further processed based on song Ids. The user's listening history is applied onto clusters via a matching algorithm to find out which cluster a song belongs to. Upon knowing the cluster, pairwise distances are calculated for all the songs within the cluster, with each of the song's attributes.

We used Big data technologies like HDFS, Spark, and HIVE database as part of our project



Clusters based on song similarity

There were many challenges that we faced while working on this project. The complete Million song data set was about 300GB and the data format was in hdf5. The data extraction from the hdf5 files was very time consuming. We had to experiment different approaches to reduce the time complexity involved in processing the million song dataset into spark understandable format. We finally processed the data successfully by running multiple jobs on HPC, parallelizing the computation. This reduced the processing time by around 80%. ☺

The spectral features of a song are 2D arrays with 12 coefficients for each segment. Each of these 12 coefficients are 12 different dimension of each spectral property of a song. The number of segments varied across all the songs. We spent considerable amount of time in researching about data preprocessing techniques, which was important for minimizing data loss on dimensionality reduction. We approached the project with the hypothesis that averages across each coefficient would standardize the data. We, therefore, normalized the data across each segment and obtained a 25 dimension vector for each song. We used this data to perform clustering of all songs on these 25 dimensions.

We executed our project in NYU HPC. There were multiple environment issues that we faced. Setting up a development environment in python was a challenge as there were many compatibility issues.

VII. FUTURE WORK

In this paper, songs are classified into clusters based on the three audio features namely pitch, timbre and loudness. This has been achieved by using the k-means algorithm.

We can expand this project by adding in additional metrics to provide better recommendations to the user. We intend to do this by performing sentiment classification of songs themselves, primarily based on lyrics and title, but some based on music analytic features as well.

Similar analysis can be conducted to build recommender system that can also be implemented on movie databases.

VIII. CONCLUSION

We divided the user playlist data into 60-40 division. We trained our analytic on the 60% data and measured the analytic on the rest of the 40% data for each user. We computed a pairwise-distance for the recommended song with the test set and we saw that our recommendation falls close to the clusters they belong to. Over all our accuracy measurement is approximately 65%

ACKNOWLEDGMENT

We would like to express our appreciation to Steve and other members of HPC who addressed our queries. We'd also like to thank our current fellow students at New York University with whom we've had many fruitful discussions on the recommender systems over this semester.

REFERENCES

[1] T. White. Hadoop: The Definitive Guide. O'Reilly Media Inc., Sebastopol, CA, May 2012.

[2] A. Gates. Programming Pig. O'Reilly Media Inc.,Sebastopol, CA, October 2011.

[3] J. Dean and S. Ghemawat. MapReduce: Simplified data processing on large clusters. In proceedings of 6th Symposium on Operating Systems Design and Implemenation, 2004.

[4] S. Ghemawat, H. Gobioff, S. T. Leung. The Google File System. In Proceedings of the nineteenth ACM Symposium on Operating Systems Principles – SOSP '03, 2003.

[5] Chen Jin, Ruogian Liu, William Hendrix, Ankit Agarwal, Alok Choudary. Scalable Hierarchial Clustering Algorithm Using Spark. IEEE First International Conference on Big Data Computing Service and Application, 2015.

[6] Brian McFee, Luke Barrington, and Gert Lanckriet IEEE. IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING, VOL. 20, NO. 8, OCTOBER 2012

[7] Caleb Jordan, Alex Cope Song Genre Claasification

[8] Yi Li, Rudhir Gupta, Yoshiyuki Nagasaki, Tianhe Zhang (Cornell University) Million Song Dataset Recomendation

[9] Using Structure Patterns of Temporal and Spectral features in audio Similarity measure