Music Recommendation System

Information Retrieval | Group - 46

Isha Sehrawat (2019046)
Srishti Singh (2019114)
Bhavesh Sood (2019355)

INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY **DELHI**

Motivation

- Growth of music industry with the advent of new technology.
- The revenue from music streaming has also been increasing every year since 2010.
- To meet the needs of these platforms, it's important to focus on improving user experience, music discovery, and data analysis.
- Music streaming platforms can enhance music discovery by providing users with better ways to find new music they will enjoy.

Problem Statement

- With the vast amount of songs in the streaming services database, it's overwhelming for users to be up to date with the newest releases and the songs matching their mood.
- While many music recommendation approaches are available, they rarely take into account the lyrics of the songs.
- We are planning to create a novel music recommendation system that takes emotions/sentiments as well as lyric features to recommend the most relevant songs and achieve user satisfaction.

Literature Review

Music recommender systems based on different dataset features have been the focus of significant research in recent years. A wide variety of features are used from the dataset ranging from audio features, user and artist metadata, and emotions associated with songs to train such systems.

Figure alongside shows what methods have been used by different researchers to work with music data.

The novelty of our idea is that we made use of sentiment pertaining to the lyrics to get a more representative embedding for the lyrics.

Model	Dataset Features
Deep Content Based [1]	Audio
Deep Multimodel [2]	Audio, Artists Biography
CAME [3]	Content and Context
HEMR [4]	Song Metadata
HRM [5]	Sequence
Ours	Lyric features, Sentiment

Novelty

- Lyrics are another crucial feature of songs that can be used to extract more information, like emotions associated with it.

- Made use of sentiment pertaining to the lyrics to get a more representative embedding for the lyrics.

Dataset Description

For our project, we are using either subset of data spawned from **The Million Song Dataset.** The three datasets we are using for our project are Song_data.csv, 10000.txt and Musixmatch dataset.

- **song_data.csv** - This is a csv file that contains song_id, title, release(album name), artist_name and year(of release).
- **10000.txt** - This dataset contains count play of 10,000 songs, including the user ID, song ID, and the number of times each song was played by the user.
- **Musixmatch Dataset** - The Musixmatch dataset consists of the song lyrics corresponding to the songs in the Million Song Dataset.

Preprocessing

Merging the two datasets

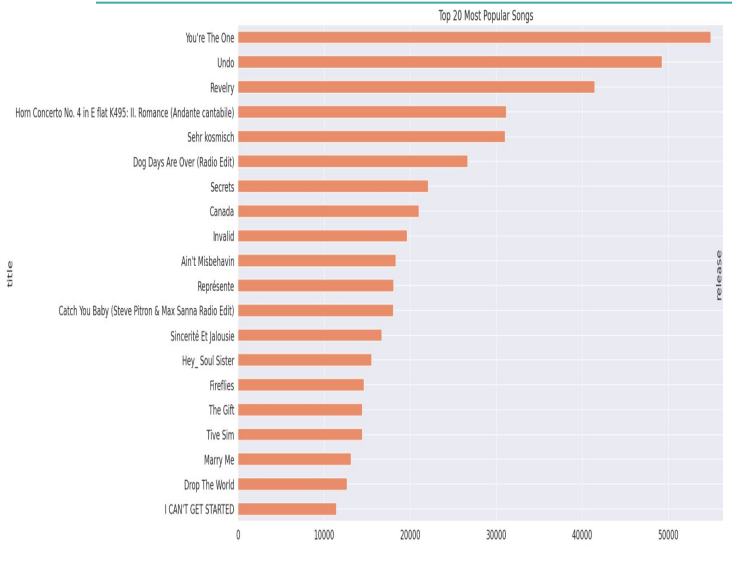
We merged the two datasets by song ID for easy access. Merging will enable us to work with song names instead of song IDs. Our code groups the song_data.csv data by song ID and selects the maximum values for each group. This effectively removes any duplicate song entries. The resulting dataframe is then merged with the 10000.txt data based on song ID. After the merge, the play_count column is renamed to listen_count, and the song_id column is dropped.

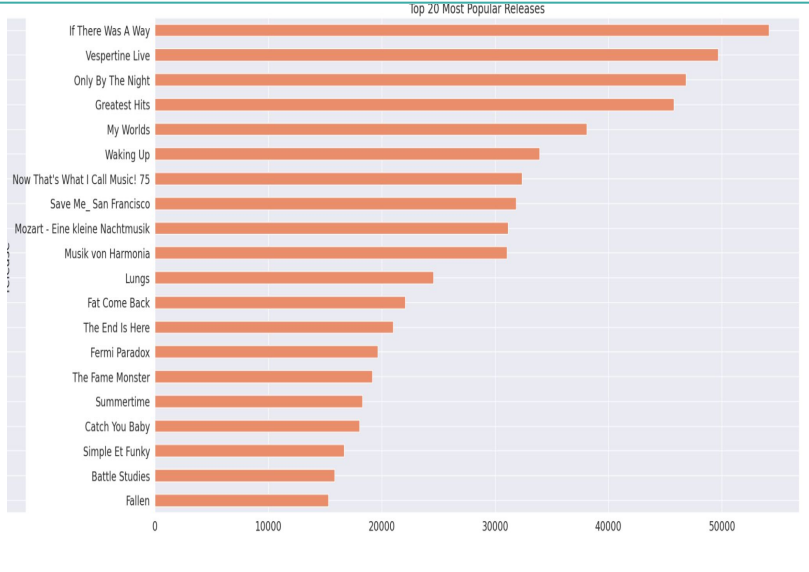
Merged Dataframe -

	user	song	listen_count	title	release	artist_name	year
0	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOAKIMP12A8C130995	1	The Cove	Thicker Than Water	Jack Johnson	0
1	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOBBMDR12A8C13253B	2	Entre Dos Aguas	Flamenco Para Niños	Paco De Lucia	1976
2	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOBXHDL12A81C204C0	1	Stronger	Graduation	Kanye West	2007
3	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SOBYHAJ12A6701BF1D	1	Constellations	In Between Dreams	Jack Johnson	2005
4	b80344d063b5ccb3212f76538f3d9e43d87dca9e	SODACBL12A8C13C273	1	Learn To Fly	There Is Nothing Left To Lose	Foo Fighters	1999
...	...	...	...	...	...	...	...
1999995	d8bfd4ec88f0f3773a9e022e3c1a0f1d3b7b6a92	SOJEYPO12AAA8C6B0E	2	Ignorance (Album Version)	Ignorance	Paramore	0
1999996	d8bfd4ec88f0f3773a9e022e3c1a0f1d3b7b6a92	SOJJYDE12AF729FC16	4	Two Is Better Than One	Love Drunk	Boys Like Girls featuring Taylor Swift	2009
1999997	d8bfd4ec88f0f3773a9e022e3c1a0f1d3b7b6a92	SOJKQSF12A6D4F5EE9	3	What I've Done (Album Version)	What I've Done	Linkin Park	2007
1999998	d8bfd4ec88f0f3773a9e022e3c1a0f1d3b7b6a92	SOJUXGA12AC961885C	1	Up	My Worlds	Justin Bieber	2010
1999999	d8bfd4ec88f0f3773a9e022e3c1a0f1d3b7b6a92	SOJYOLS12A8C13C06F	1	Soil_ Soil (Album Version)	The Con	Tegan And Sara	2007

2000000 rows × 7 columns

Exploratory Data Analysis





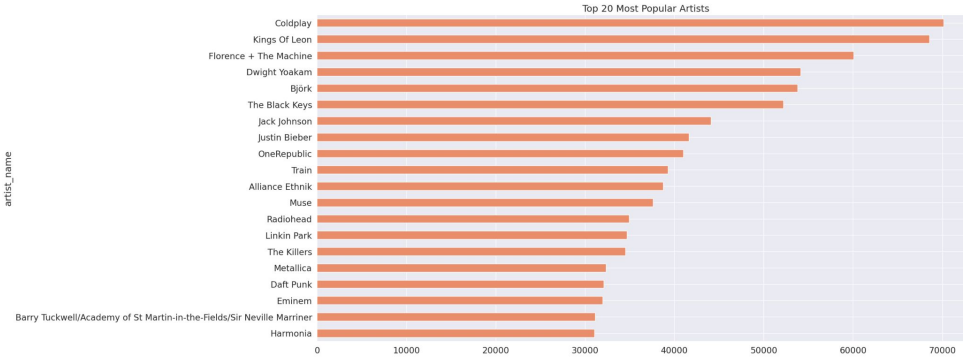
	listen_count
count	2.000000e+06
mean	3.045485e+00
std	6.579720e+00
min	1.000000e+00
25%	1.000000e+00
50%	1.000000e+00
75%	3.000000e+00
max	2.213000e+03

- We observe that there was a song which was played 2213 times by a single user

Popularity Based

The most simplest recommendation engine is the ***Popularity-Based***, that basically standing, if some item is liked by a vast majority of our user base, then it is a good idea to recommend that item to users who have not interacted with that item.

	title	score	rank
6837	Sehr kosmisch	8277	1.0
8726	Undo	7032	2.0
1965	Dog Days Are Over (Radio Edit)	6949	3.0
9497	You're The One	6729	4.0
6499	Revelry	6145	5.0
6826	Secrets	5841	6.0
3438	Horn Concerto No. 4 in E flat K495: II. Romanc...	5385	7.0
2596	Fireflies	4795	8.0
3323	Hey_ Soul Sister	4758	9.0
8495	Tive Sim	4548	10.0
8781	Use Somebody	3976	11.0
5721	OMG	3947	12.0
2120	Drop The World	3879	13.0
5000	Marry Me	3578	14.0
1265	Canada	3526	15.0



Item Similarity Based

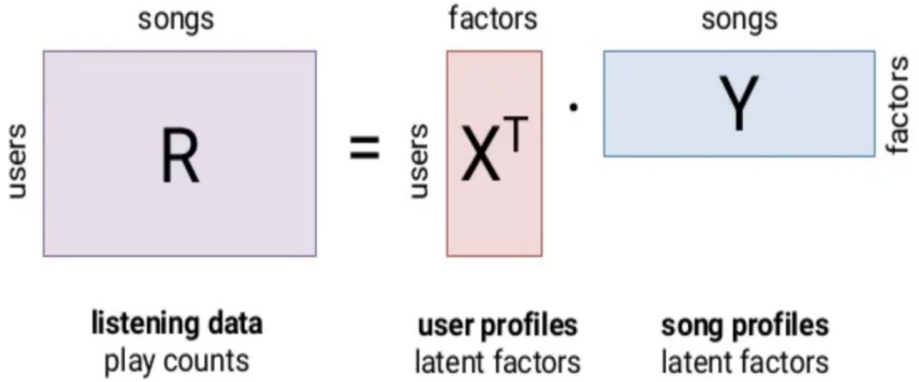
- This method focuses on the similarity between the user's list of songs (listening history, their playlists, etc.) and song data for other users that are present in the training data.
- Calculates the similarity between the user list of songs and songs in our dataset using the Jaccard index based on common users of songs.
- This method provides more personalized recommendations based on the user's listening history and preferences, thus improving the overall user experience and engagement.
- We compared the results by listening to the top song in the recommendation list and the song most listened to by the user.

Recommendation list by Item Similarity Based Engine-

	user_id	song	score	rank
0	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Quiet Houses	0.044710	1
1	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Meadowlarks	0.043836	2
2	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Heard Them Stirring	0.042740	3
3	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Tiger Mountain Peasant Song	0.041485	4
4	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Sun It Rises	0.040973	5
5	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Your Protector	0.039942	6
6	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Oliver James	0.039287	7
7	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Great Indoors	0.036765	8
8	b80344d063b5ccb3212f76538f3d9e43d87dca9e	White Winter Hymnal	0.036345	9
9	b80344d063b5ccb3212f76538f3d9e43d87dca9e	If I Could	0.034576	10

Latent Factor Model

- Matrix factorization is a powerful technique used in machine learning to discover latent features between two different kinds of entities
- The algorithm aims to minimize the reconstruction error between the observed listen counts and the predicted listen counts obtained by the product of the user and songs latent factor matrices.
- Assuming the process helps us identify latent factors/features, meaning as K, our aim is to find two matrices X and Y such that their product (matrix multiplication) approximates R.
- X = |U| x K matrix (A matrix with dimensions of num_users * factors)
- Y = |P| x K matrix (A matrix with dimensions of factors * num_songs)



15

Sentiment Analysis Model using Lyrics

- Sentiment Analysis used to extract emotional tone of a piece of text
- Titles of songs suggest the emotional tone of those songs for majority of the cases
- We used Vader Sentiment Library to extract sentiment scores associated with different song titles
- Lyrics provide valuable information about the content and style of a song, which can influence a user's musical preference
- They can be analyzed to determine the emotional tone of a song

New Dataset with Lyrics

- Merged dataset does not contain song lyrics
- Used the musixmatch dataset available in the MSD
- Mapped MSD song IDs with the Million Song Dataset Song IDs to resolve song lyrics
- Lyrics present in bag of words format
- Used FastText model to get non contextual embedding for the lyrics
- FastText well suited for bag of words format
- Song Titles embedded in a similar format
- All lyrics and song title converted to 300 length vector embeddings
- Final dataset after all processing can be found here.

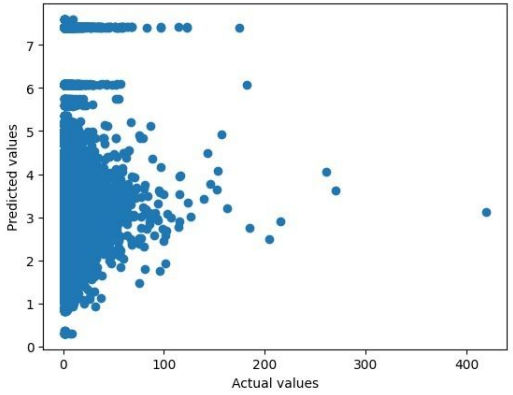
	3	9	10	11	12	13	14	15	16	17	...	602	603	604	605	606	607	608	609	610	listen count
0	36168	1976	0.013271	-0.045020	-0.008184	0.049428	-0.025430	-0.008316	-0.042515	-0.007629	...	0.051514	0.112377	-0.101461	-0.009572	0.013096	-0.002825	0.099568	-0.025742	0.019022	2
1	36168	2007	-0.016372	-0.012811	0.019531	0.062848	-0.048744	-0.004653	0.018181	0.012372	...	0.033598	0.061161	-0.149784	0.123100	0.025756	0.042525	0.082381	0.098697	0.044474	1
2	36168	2005	0.012538	-0.005735	0.014831	0.052208	-0.050455	-0.028510	-0.014352	-0.001575	...	-0.082567	0.082334	-0.025078	0.001777	-0.088783	-0.084012	0.079264	-0.034398	0.011239	1
3	36168	1999	0.006611	-0.003212	0.007064	0.051910	-0.043352	-0.016626	-0.010534	0.005963	...	-0.024937	0.020670	-0.038837	-0.064640	-0.012014	-0.012017	0.038513	0.001047	-0.035434	1
4	36168	0	-0.007878	-0.010194	-0.004973	0.053964	-0.026041	0.001560	-0.028458	-0.002844	...	-0.009999	0.110747	0.005159	-0.012548	-0.073592	0.056958	0.083830	-0.025329	0.052146	1

Models And Results

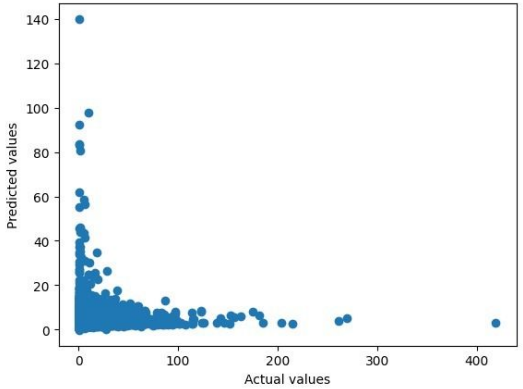
Model	**MSE**	**R^2 Error**
Linear Regression	35.66	0.02
XGBoost	36.61	-0.01
Deep Learning based	36.21	0.00

ML– Models

- We observe that the simplest model Linear Regression gave a better MSE and R^2 , this must be due to the huge data, and also that a line fits well since most values lie between 1 to 10.



Linear Regression

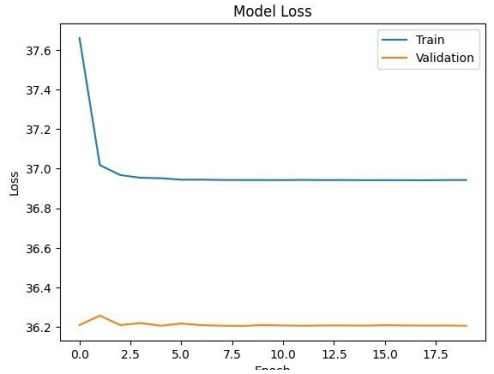


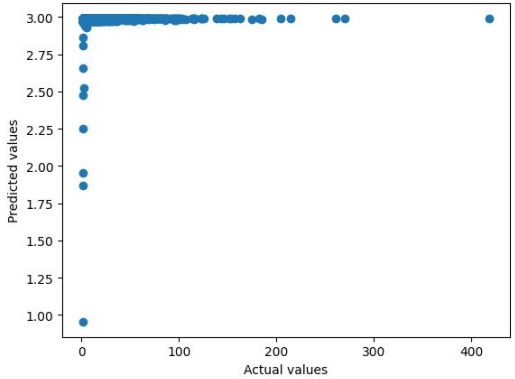
XGBoost

DL Model

- We made this model using a [Dense -> BatchNorm -> Dropout] block strategy going down the dimensions.
- The error however didn't decrease very much as compared to the ML models, because
  - the model is underfit, as it was a large dataset it took a long time to train.
  - we see that the model somehow made a max limit on prediction at 3.

```
model = Sequential()
model.add(Dense(256, input_dim=602, activation='relu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))
model.add(Dense(128, activation='relu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))
model.add(Dense(64, activation='relu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))
model.add(Dense(32, activation='relu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))
model.add(Dense(16, activation='relu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))
model.add(Dense(1, activation='linear'))
```





Thank You !!