

# Music search – Million Playlist Challenge data

Team Python

Chirag Bhatt 2018MT10750

Parth Singhal 2018CH10231

Rohan Sharma 2018TT10909

## 1 TASK

The Spotify Million Playlist Dataset Challenge consists of a dataset and evaluation to enable research in music recommendations. We are provided with a dataset consisting of 1 million playlists containing 2262292 unique tracks from 734684 unique albums. The goal of the challenge is to develop a system for the task of automatic playlist continuation. Given a set of playlist features, participants' systems shall generate a list of recommended tracks that can be added to that playlist.

We define the task formally as follows:

Input : A user-created playlist, represented by:

1. Playlist metadata (name, number of tracks, last modified etc.)
2. a list of K tracks in the playlist, K can be equal to 0, 1, 5, 10, 25, or 100.

Output : A list of 500 recommended candidate tracks, ordered by relevance in decreasing order.

Track is characterized by track name, track id, artist name, artist id, album name, album id and duration while a playlist is characterized by the playlist name and the tracks it contains.

## 2 EVALUATION

Submissions will be evaluated using the following metrics. All metrics will be evaluated at both the track level (exact track match) and the artist level (any track by the same artist is a match). In the following, we denote the ground truth set of tracks by G and the ordered list of recommended tracks by R

1. R- precision : R-precision is the number of retrieved relevant tracks divided by the number of known relevant tracks (i.e., the number of withheld tracks) The metric is averaged across all playlists in the challenge set. This metric rewards the total number of retrieved relevant tracks (regardless of order).  
$$\text{R-precision} = |\text{G} \cap \text{R}_{1:|\text{G}|}|/|\text{G}|$$
2. Normalized Discounted Cumulative Gain : Discounted Cumulative Gain (DCG) measures the ranking quality of the recommended tracks, increasing when relevant tracks are placed higher in the list. Normalized DCG (NDCG) is determined by calculating the DCG and dividing it by the ideal DCG in which the recommended tracks are perfectly ranked  
$$\text{DCG} = \text{rel}_1 + \sum_{i=2}^G \text{rel}_i / \log_2 i$$
$$\text{Ideal DCG or IDCG} = 1 + \sum_{i=2}^{|\text{R} \cap \text{G}|} 1 / \log_2 i$$
$$\text{NDCG} = \text{DCG} / \text{IDCG}$$
3. Recommended songs clicks : It is a spotify feature that, given a set of tracks in a playlist, recommends 10 tracks to add to the playlist. The list can be refreshed to produce 10 more tracks. Recommended Songs clicks is the number of refreshes needed before a relevant track is encountered. If the metric does not exist (i.e. if there are no relevant tracks in R, a value of 51 is picked (which is 1 greater than the maximum number of clicks possible).  
$$\text{Recommended songs clicks} = \lfloor (\arg \min_i \{ \text{R}_i : \text{R}_i \in \text{G} \} - 1) / 10 \rfloor$$

As part of the challenge, a separate challenge dataset ("test set") that consists of 10,000 playlists is given with incomplete information. More specifically, the challenge dataset is divided into 10 scenarios, with 1000 examples of each scenario:

- (a) Title only (no tracks)
- (b) Title and first track
- (c) Title and first 5 tracks
- (d) First 5 tracks only
- (e) Title and first 10 tracks
- (f) First 10 tracks only
- (g) Title and first 25 tracks
- (h) Title and 25 random tracks
- (i) Title and first 100 tracks
- (j) Title and 100 random tracks

### **3 PLAN AHEAD**

We have some ideas from which we would like to begin with. Our initial idea is to calculate relevance score of tracks based on four factors that are similarity in the artist, album, name of the track and name of the associated playlist and then experimentally find out relative weightage of different factors. We would like to begin with this approach and continue/modify/change based on the results we get. By taking the reference from relevant research paper, we also plan to implement collaborative filtering and cold start problem based models

### **4 SOURCE**

<https://www.aicrowd.com/challenges/spotify-million-playlist-dataset-challenge>

### **5 RELEVANT RESEARCH PAPER**

[https://www.researchgate.net/publication/328086857\\_Artist-driven\\_layering\\_and\\_user's\\_behaviour\\_impact\\_on\\_recommendations\\_in\\_a\\_playlist\\_continuation\\_scenario](https://www.researchgate.net/publication/328086857_Artist-driven_layering_and_user's_behaviour_impact_on_recommendations_in_a_playlist_continuation_scenario)