Automated Playlist Generation

Kade Keith, Demetrios Fassois
{kade, dimifass}@stanford.edu
CS 229, Fall 2017

MOTIVATION

With the growth of music streaming services, there are now more songs than ever at music listeners fingertips. Because of this growth, the art of constructing playlists has become increasingly challenging, and discovering new music the in the expanse of choices is a daunting task. We explore multiple approaches to, given some set of songs as a seed, deciding which other songs belong on the playlists based on lyrical and audio features. These models are trained on and evaluated against human-generated playlists.

DATA & FEATURES

Our data comes from the Million Song Dataset and musiXmatch lyrics collection. We augment those songs with info from Spotify. Our playlists are from Spotify.

Raw features:

- Tempo, Timbre, Danceability, Energy, Loudness, Speechiness, Acousticness, Instrumentalness, Liveness, Valence (Audio Features)
- Popularity, Year (Song Metadata)
- Lyrics in bag-of-words format

Derived features:

- TODO (HMM path value)
- Lyrics category (determined via LDA)
- Lyrical sentiment (determined via Naive Bayes)

We chose our features based on what we found important while making our own playlists, and thus what we believe is important to others.

MODELS

Classification-based

Our primary approach was to think of selecting songs for a playlist as a classificaion problem.

Logistic Regression

Lorem ipsum dolor sit amet, mutat quodsi usu ei, ne quis nullam eruditi pro. Doming menandri ex sea, mel et sonet corpora phaedrum, per at veri porro neglegentur.

TODO

Lorem ipsum dolor sit amet, mutat quodsi usu ei, ne quis nullam eruditi pro. Doming menandri ex sea, mel et sonet corpora phaedrum, per at veri porro neglegentur.

Graph-based

An interesting alternative is to think of songs as nodes on a completely connected graph.

k-Nearest Neighbors

We represent each song as a point in n-dimensional space according to our normalized features. Then we select the next songs for that playlist based on proximity to the seed.

Shortest path (of length k)

Given two songs, we construct a playlist as the shortest path of length k between them. This is particularly intriguing if the two seed songs are very different. Via dynamic programming, the shortest path to node v of length k is:

$$path[v][k] = \min_u \left(path[u][k-1] + dist[u][v] \right)$$

RESULTS

These results are averages of 41 popular playlists that we we split individually into 75/25 test/train sets. These playlists had an average of 46 songs each (that we had data on). We used an equal number of randomly samped of songs not on the playlist for negative examples.

Classification	Train error	Test error
SVM		
Logistic Regression		
TODO		

For the classification approach, we report train and test error. For KNN approach we report average normalized distance within the playlists, and betwen the predictions and the positive train data, positive test data, and a random sampling of songs. For the shortest path, we construct a path between two arbitrarily chosen songs from the playlist, and report the distance to the actual playlist.

	Distance within playlists	Distance to train data (KNN)	Distance to test data (KNN)	Distance to random (KNN)	Distance to playlist (shortest-path)
Graph Approaches	0.217	0.172	0.173	0.257	0.193

DISCUSSION

We are pleased with how both approaches to the task performed, and we attribute the accuracy to the strength of our features, both raw and derived. In particular we saw increased performance when adding popularity and lyrics data. When making a playlists, people choose songs that make them feel a certain way, and lyrics are huge part of that. Also people choose songs that they've actually heard of, so popularity is important. Our training data was biased towards popular songs in particular because we chose top results.

FUTURE

For starters, we would gather even more data - particularly in underrepresented genres in the MSD like Hip Hop. Beyond that we think it would be really interesting to compare the learned parameters of our various models between playlists. This way we can examine what kind of differences there are in what motivates individuals when they make playlists.

REFERENCES

1. Million song dataset. https://labrosa.ee.columbia.edu/millionsong/.
2. Spotify. https://www.spotify.com/.
3. Beth Logan. Content-Based Playlist Generation: Exploratory Experiments
4. Masoud Alghoniemy and Ahmed H. Tewfik. A network flow model for playlist generation. In In Proc IEEE Intl Conf Multimedia and Expo, 2001.