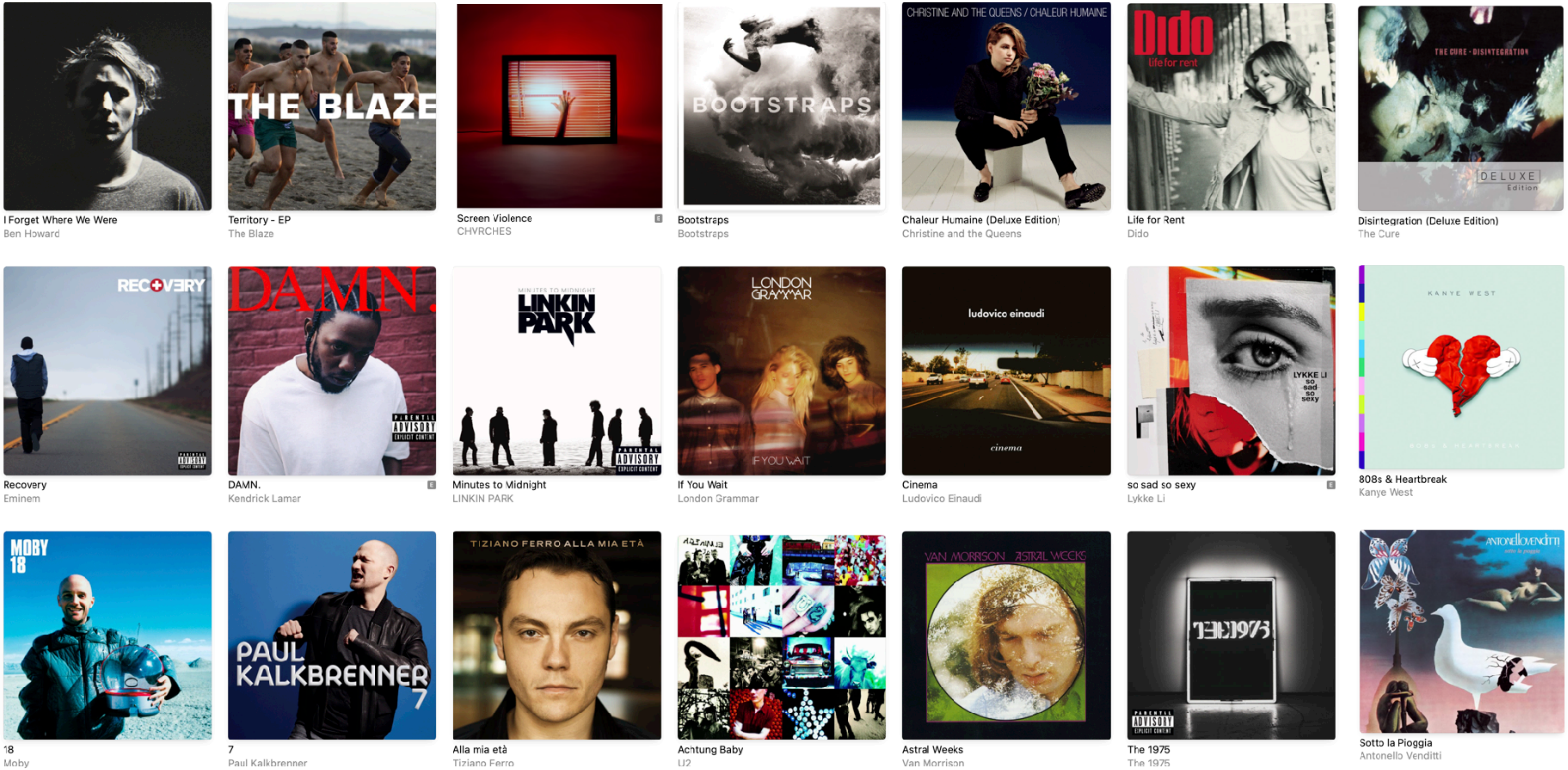


Recommended Frequencies

A Music Recommendation Engine
by Lucas Chizzali

Listen to music

Analyse music

Recommended Frequencies
What?

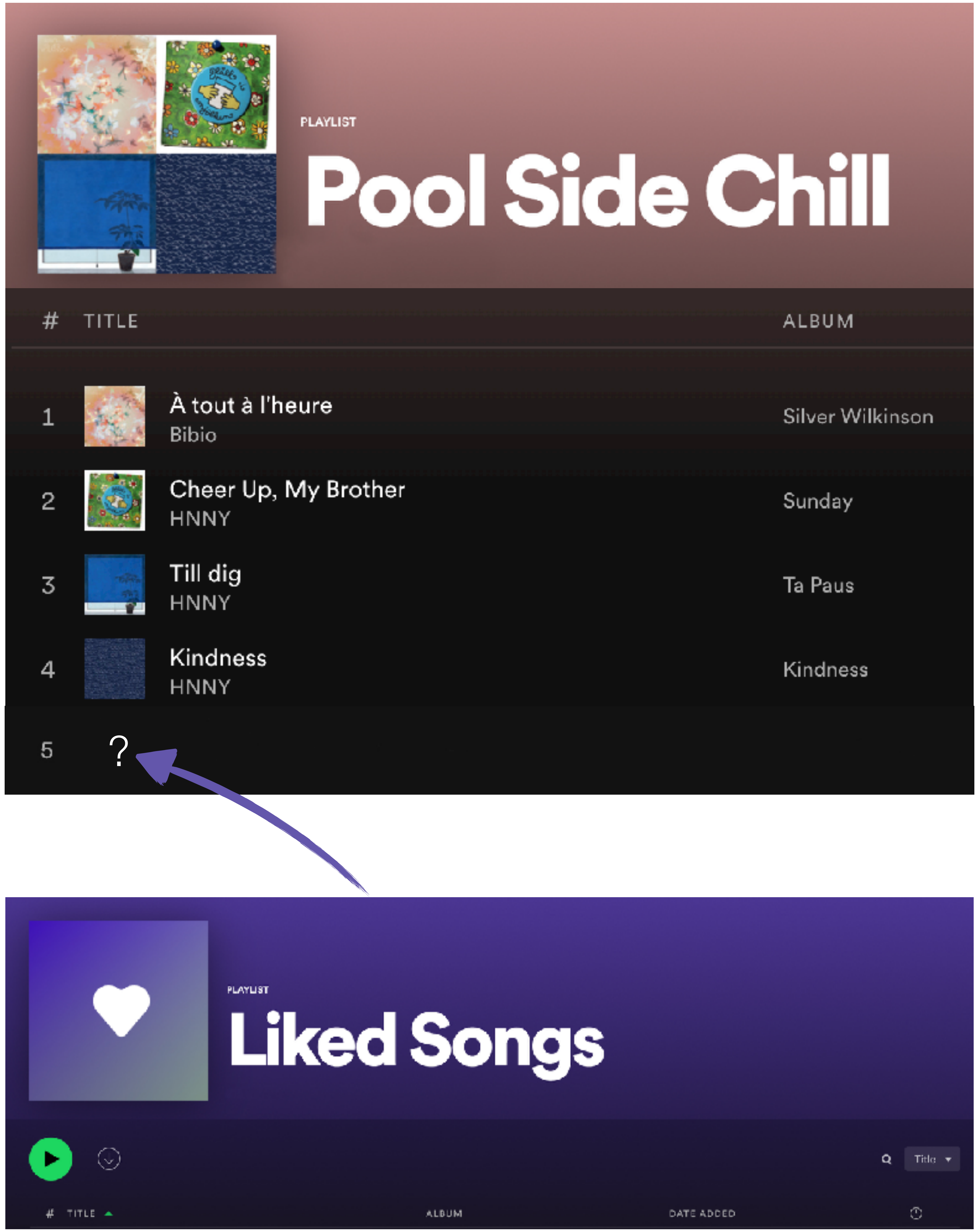
Help manage playlists

→ Identify songs in a person's music library that may fit well into a chosen, existing playlist

Multifaceted and difficult task since music is subjective

Specifically, a playlist may represent

- Mood (Summertime 🍉)
- Memory (High School 🎓)
- Genre (Synthwave)
- Era (80s 📼)
- etc.



Recommended Frequencies
Why?

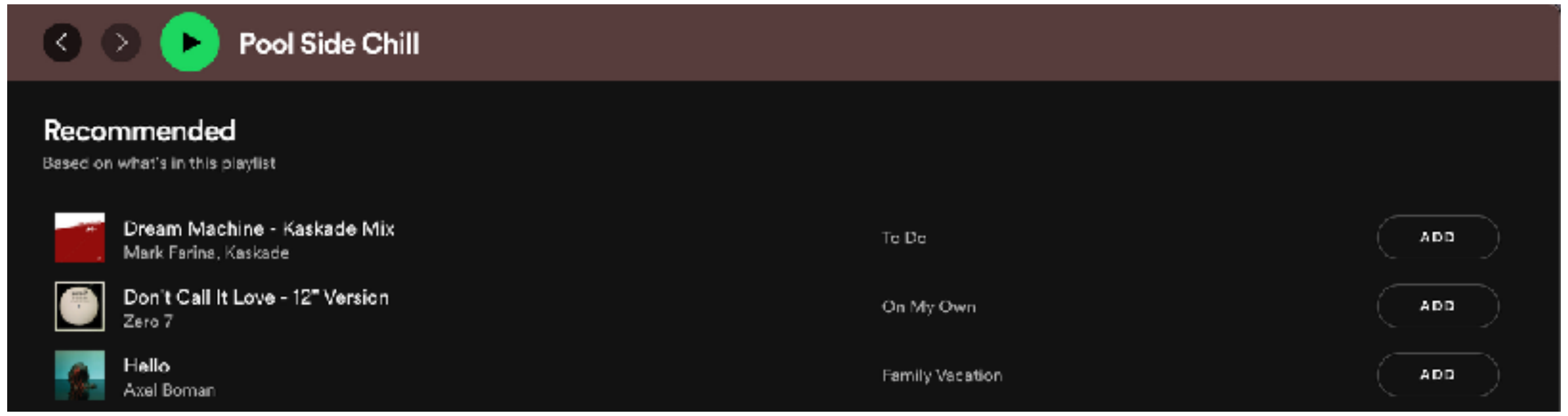
Playlists are an integral means to enjoying music

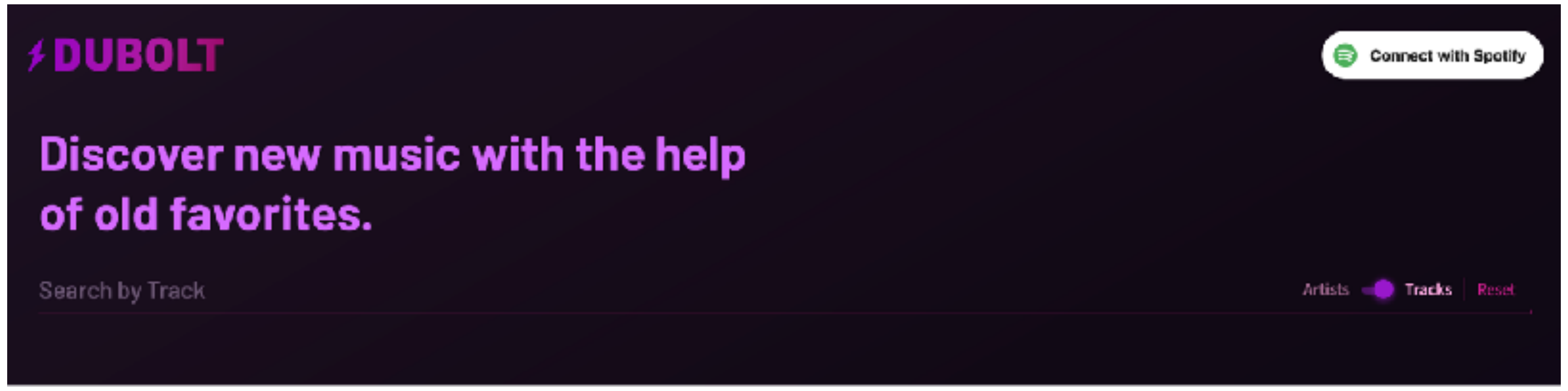
- Anecdotal evidence
- 4 bn playlists vs 350 m users on Spotify*

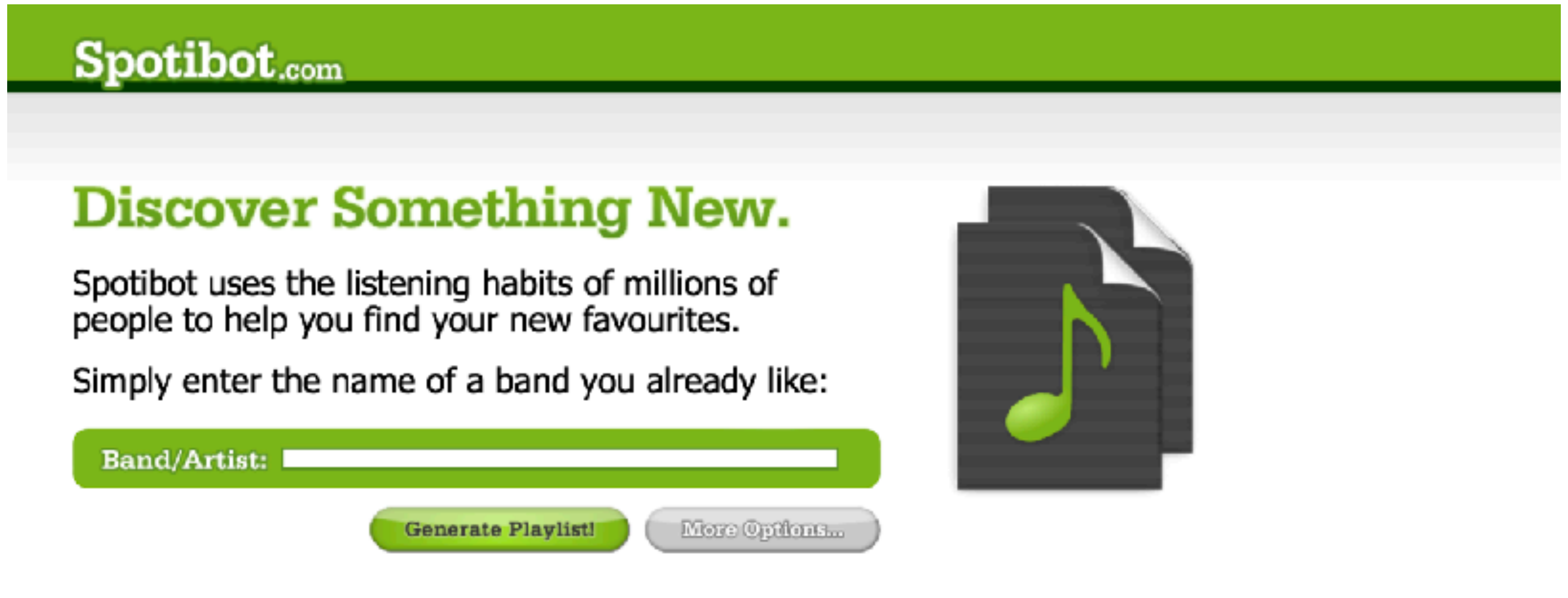
Music recommendation is nothing new

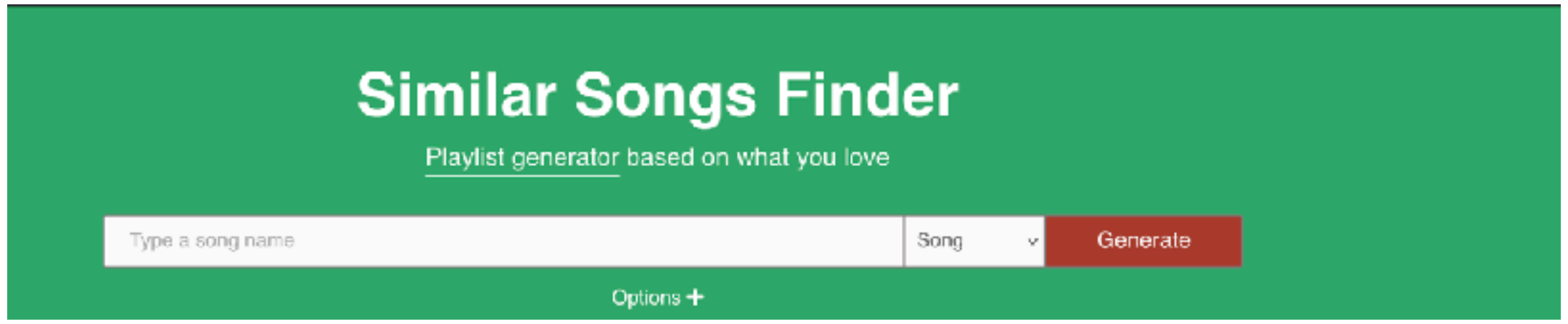
Nevertheless

- DIY is fun
- Re-discovering your own music









*https://soundplate.com/how-many-playlists-are-there-on-spotify-and-other-spotify-stats/

Let's research the

Music Recommendation
How?

- Popular researched domain
- Competitive edge for streaming services
- Usage ***vs*** content data

	Usage data	Content data
Approaches	Recommender Systems	Music Information Retrieval
Example of technique	Collaborative filtering [1]	Deep Learning [2]
Example of datasource		Million Song Dataset

[1] Song, Yading, Simon Dixon, and Marcus Pearce. "A survey of music recommendation systems and future perspectives." In 9th international symposium on computer music modeling and retrieval, 2012.
[2] Van den Oord, Aaron, Sander Dieleman, and Benjamin Schrauwen. "Deep content-based music recommendation." *Advances in neural information processing systems* 26 (2013)

Music Recommendation
Implications

	Usage data	Content data
Approaches	Recommender Systems	Music Information Retrieval
Example of technique	Collaborative filtering [1]	Deep Learning [2]
Example of datasource		Million Song Dataset

Usage data

- Not easy to come by
- Proprietary / limited access to the public
- Large amounts of data

Content data

- Limited number of songs available
- Pre-trained models for audio → features
- Audio not straightforward to come by (e.g. copyright)

For now, let's devise a

- Only use information relating to a person's music library
- Work with readily available audio features
- Quickly build an easy-to-use tool to consume recommendations

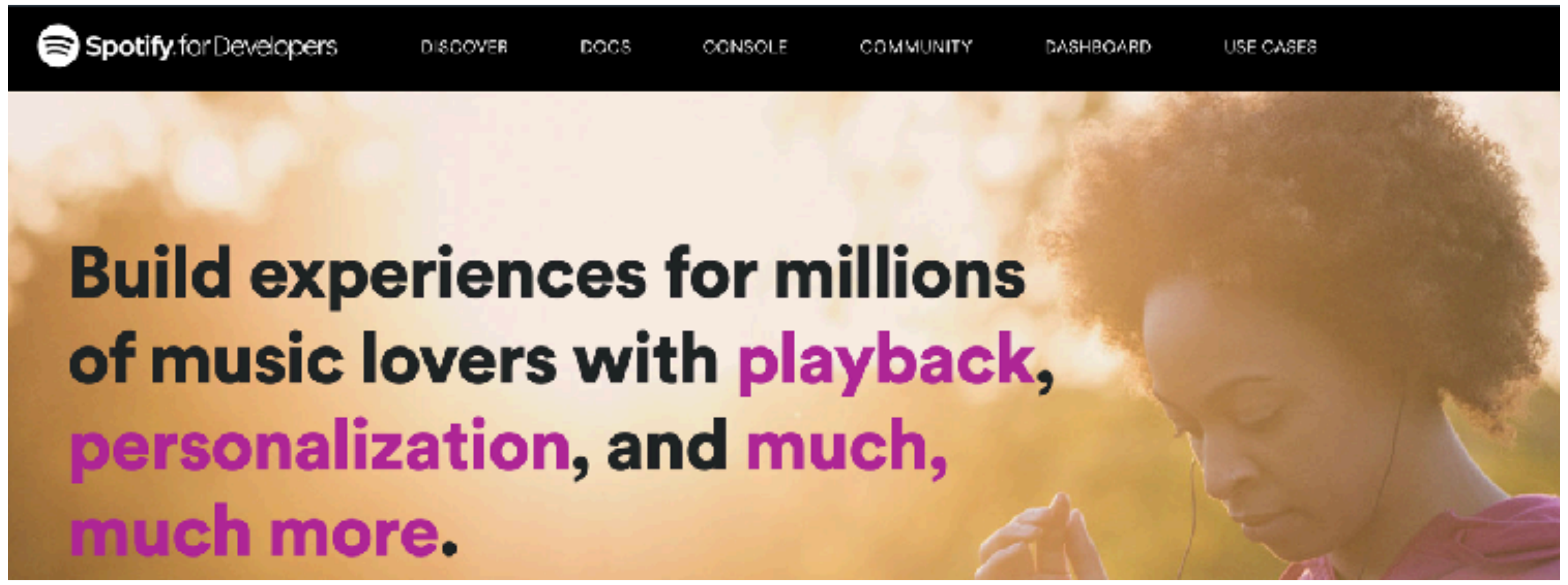
Recommended Frequencies

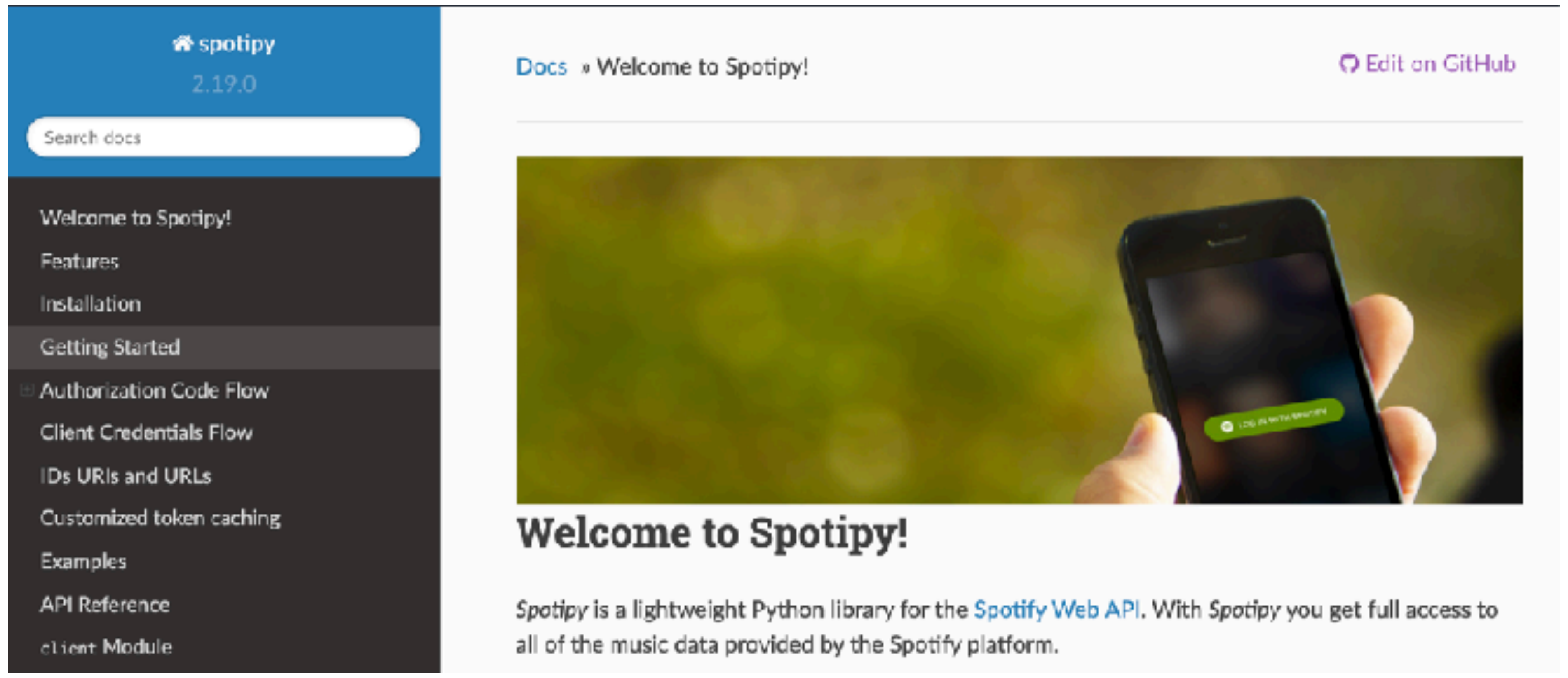
How?

Use readily available audio features

Spotify API

- Free
- Popular
- Audio features + metadata
- Python library

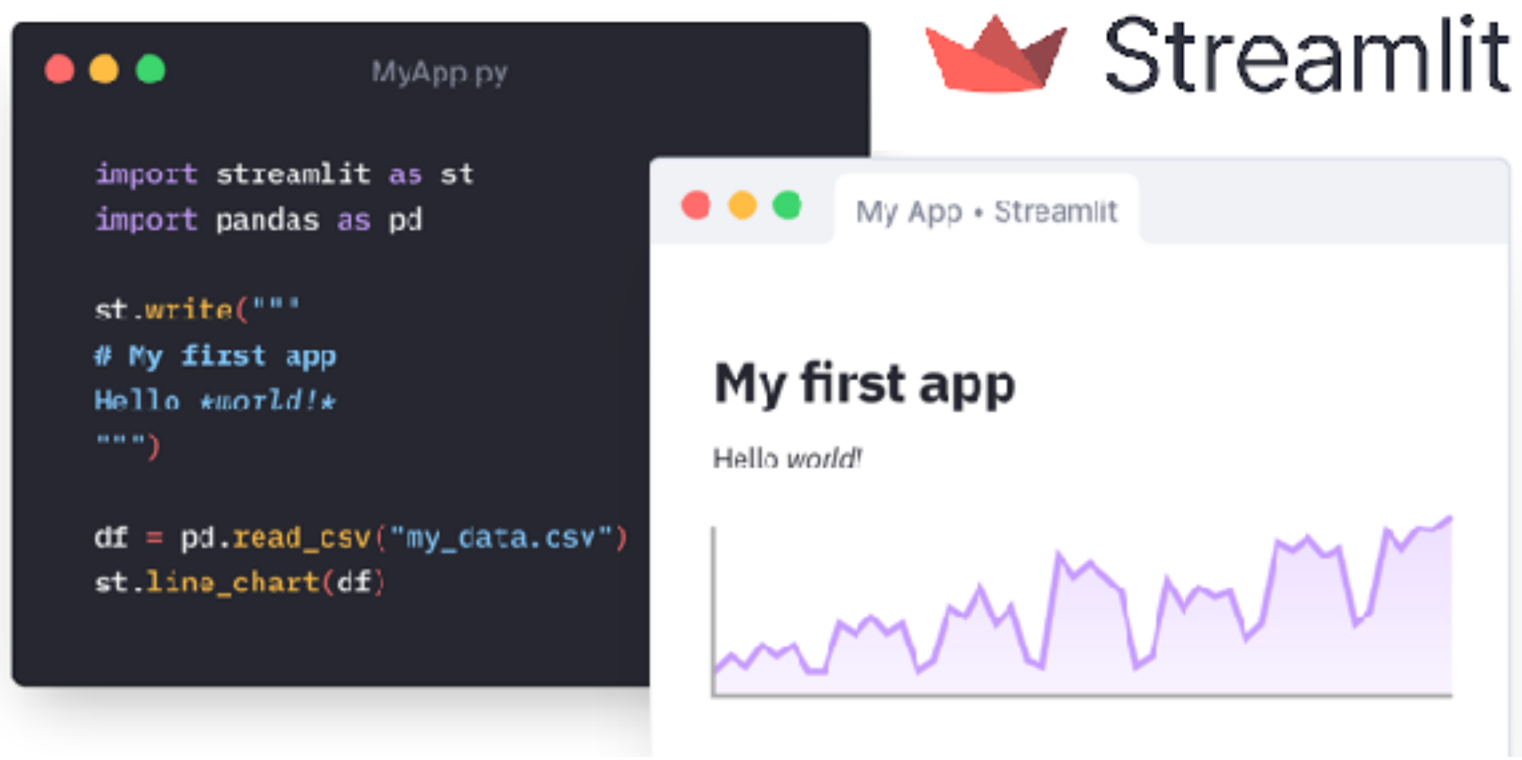




Build a dashboard for interacting with results

Streamlit

- "The fastest way to build and share data apps"



Data, or should I say…

Spotify API

Audio Features

Body application/json

audio_features array of objects required

acousticness number<float>

A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.

>= 0 <= 1

danceability number<float>

Danceability describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.

energy number<float>

Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale. Perceptual features contributing to this attribute include dynamic range, perceived loudness, timbre, onset rate, and general entropy.

instrumentalness number<float>

Predicts whether a track contains no vocals. "Ooh" and "aah" sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly "vocal". The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0.

liveness number<float>

Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live.

loudness number<float>

The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks. Loudness is the quality of a sound that is the primary psychological correlate of physical strength (amplitude). Values typically range between -60 and 0 db.

speechiness number<float>

Speechiness detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks.

tempo number<float>

The overall estimated tempo of a track in beats per minute (BPM). In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration.

valence number<float>

A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry).

>= 0 <= 1

Spotify API
Song Metadata

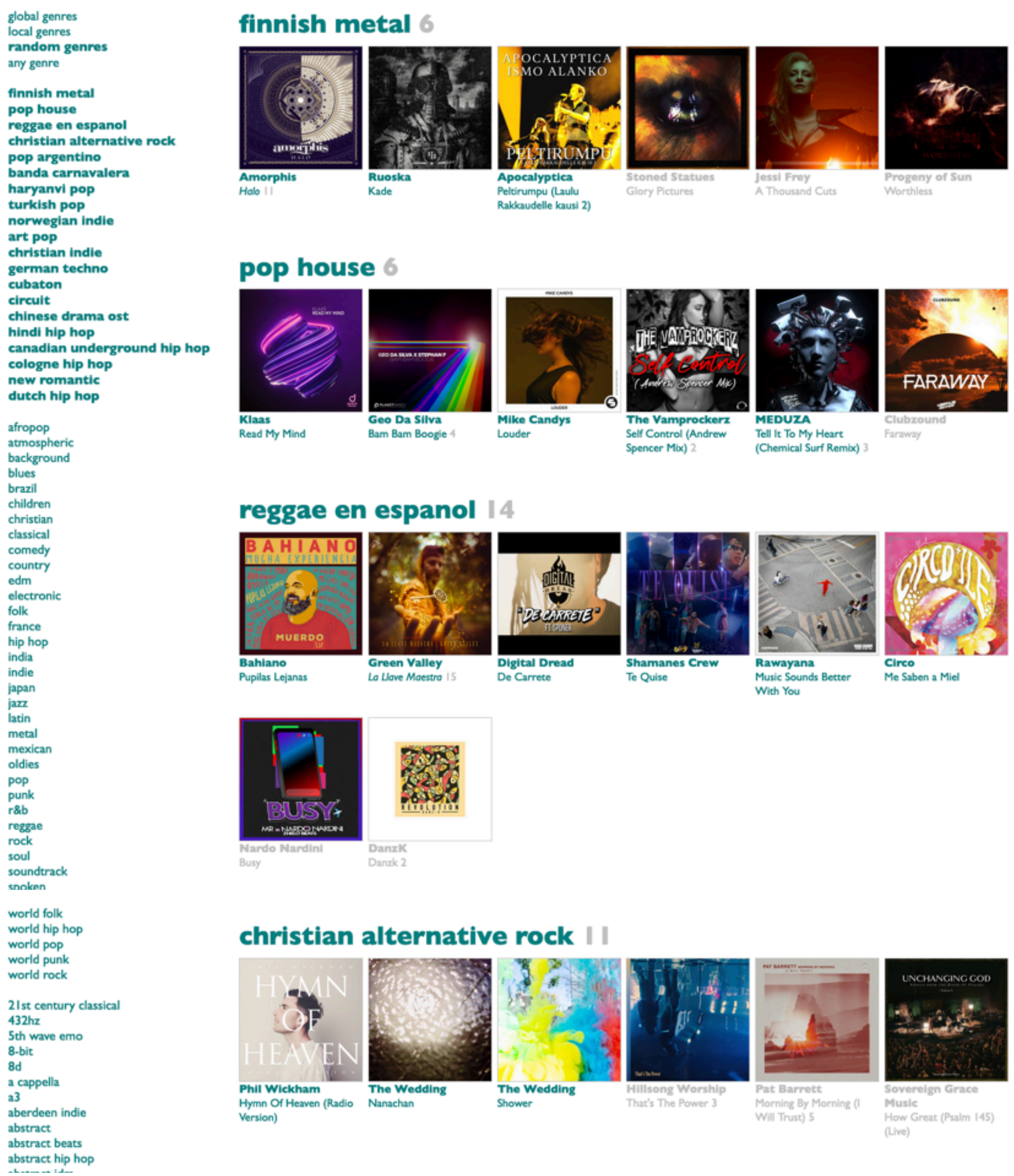
release_date string — required

The date the album was first released.

genres array of strings

A list of the genres the artist is associated with. If not yet classified, the array is empty.

A lot of genres exist (>5k)!



https://everynoise.com/new_releases_by_genre

Methodology, a.k.a

Audio Feature Similarity

Weighted Euclidean distance

$$d(\boldsymbol{P}, \boldsymbol{s}) = \sqrt{\sum_{i}^{d} \log\left(\frac{1}{\sigma(\boldsymbol{P}_i)}\right)(\mu(\boldsymbol{P}_i) - \boldsymbol{s}_i)^2}$$

Average of feature across songs in playlist

Weigh each feature by its standard deviation across songs in the playlist

- Inverse: more uniform → more important
- Log: limit influence of highly uniform features

where

- feature vector of a song $\boldsymbol{s} \in \mathbb{R}^d$
- feature matrix of a playlist $\boldsymbol{P} = \begin{bmatrix} \boldsymbol{s_1} \\ \vdots \\ \boldsymbol{s_n} \end{bmatrix} \in \mathbb{R}^{n \times d}$
- d = 9 audio features + release date

Look at this graph…

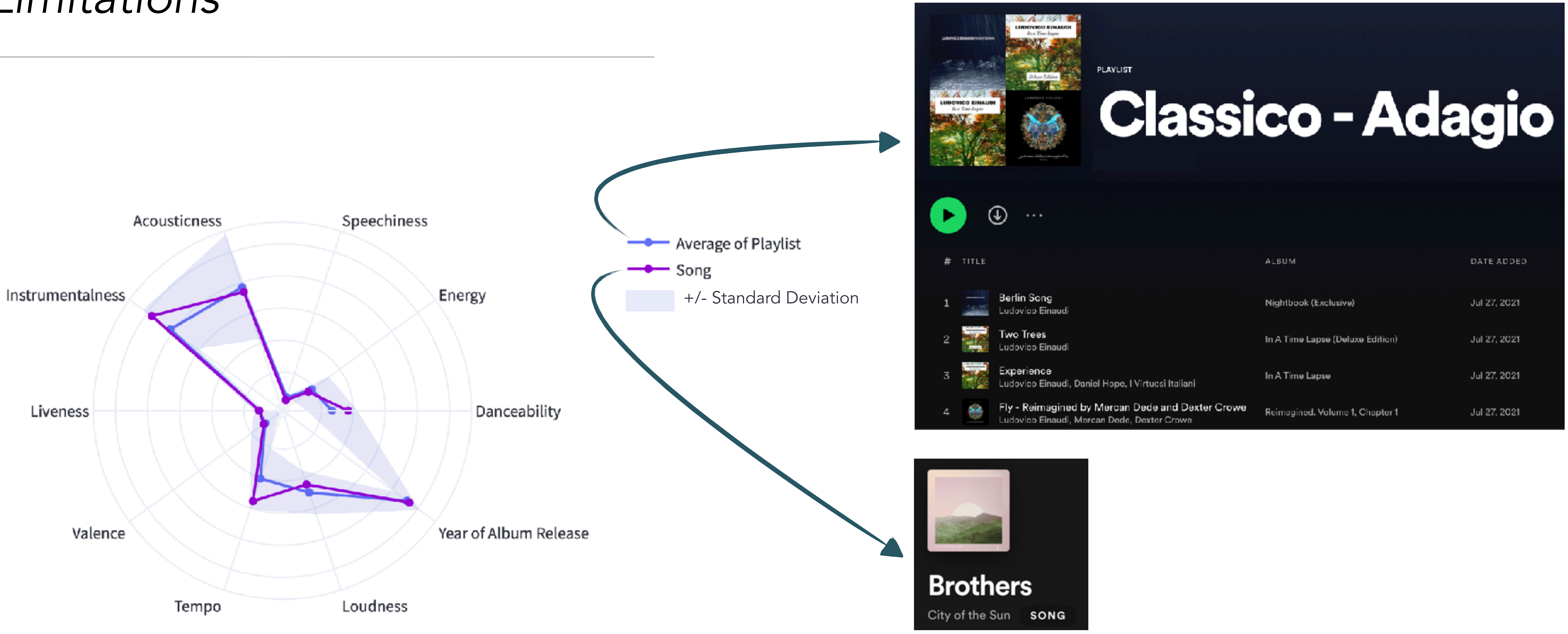
… every time it makes me laugh

Audio Feature Similarity
Visualisation

Similarity for this feature is more important since majority of songs share the same attribute

Similarity for this feature is less important since few songs share the same attribute

Audio Feature Similarity
Limitations



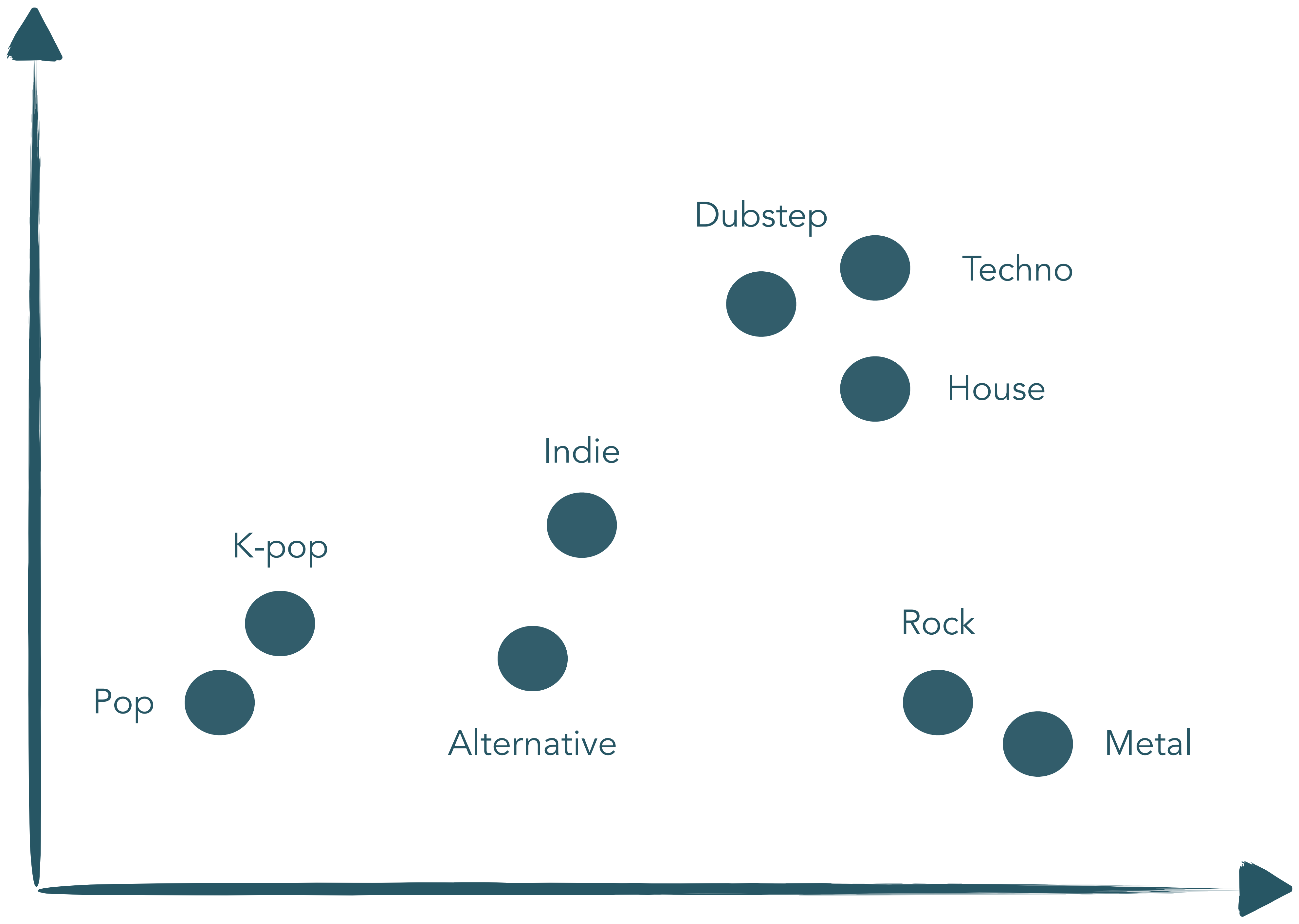
Genre Similarity

Many options

1. Set similarity
2. Co-occurrence
3. Word2Vec embedding
4. "Every Noise at Once" embedding

and more

Similar genres should be close together



Illustration

Genre Similarity
Jaccard index & co-occurrence

Jaccard index

Genres in playlist

Genres of song

$$\mathrm{sim}(g_P, g_s) = \frac{|g_P \cap g_s|}{|g_P \cup g_s|}$$

Co-occurrence

$$\text{co-occurrence}(g_1, g_2) = \sum_i f(g_1, g_2, P_i)$$

$$\text{where} \quad f(g_1, g_2, P_i) = \begin{cases} 1, & \text{if } (g_1 \in P_i) \cap (g_2 \in P_i) \\ 0, & \text{otherwise} \end{cases}$$

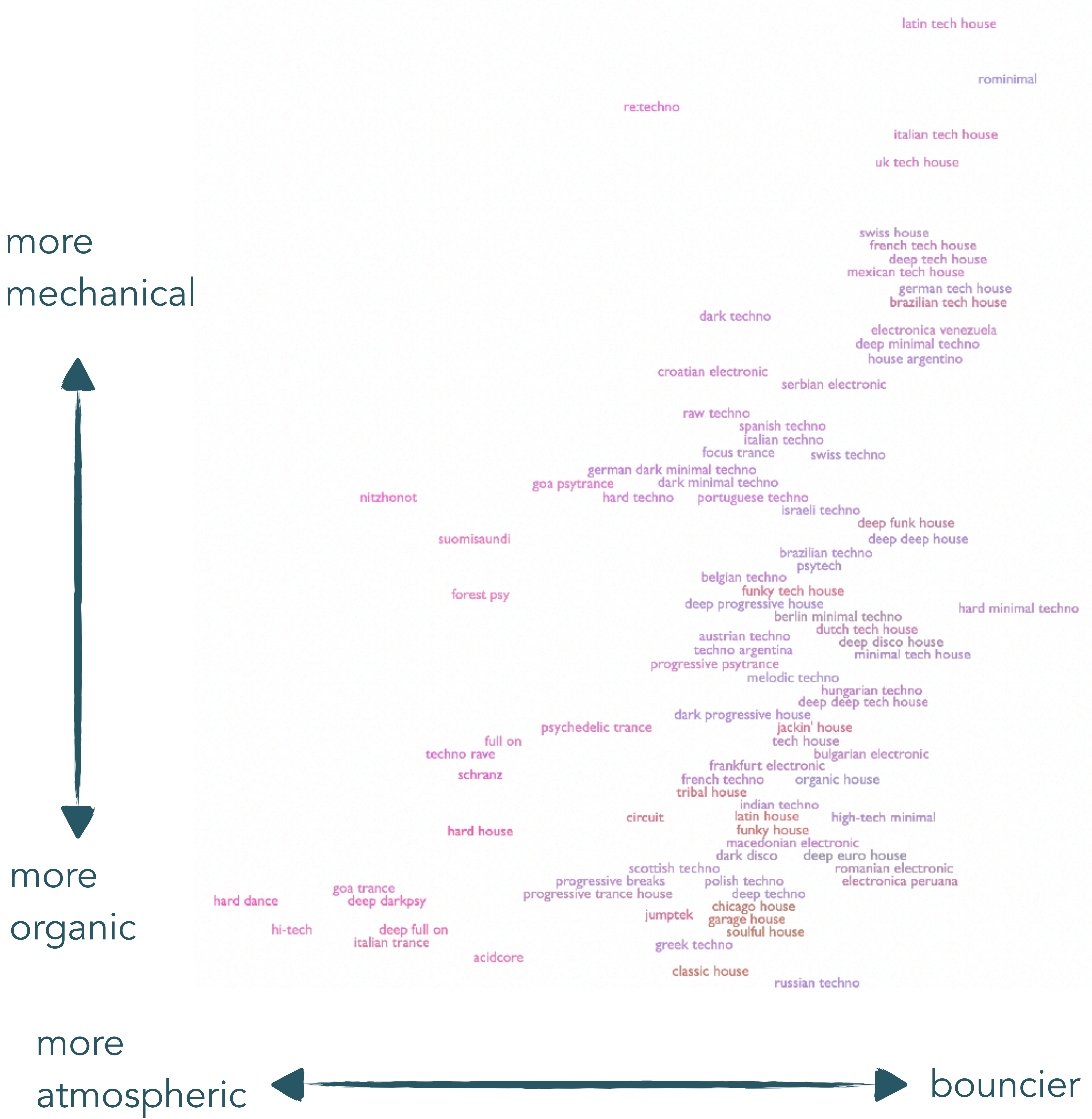
Playlist

Genre Similarity
Every Noise at Once

- everynoise.com provides an "algorithmically-generated […] scatter-plot of the musical genre-space" of Spotify
- Embedding equals (x, y) coordinates of genre in plot

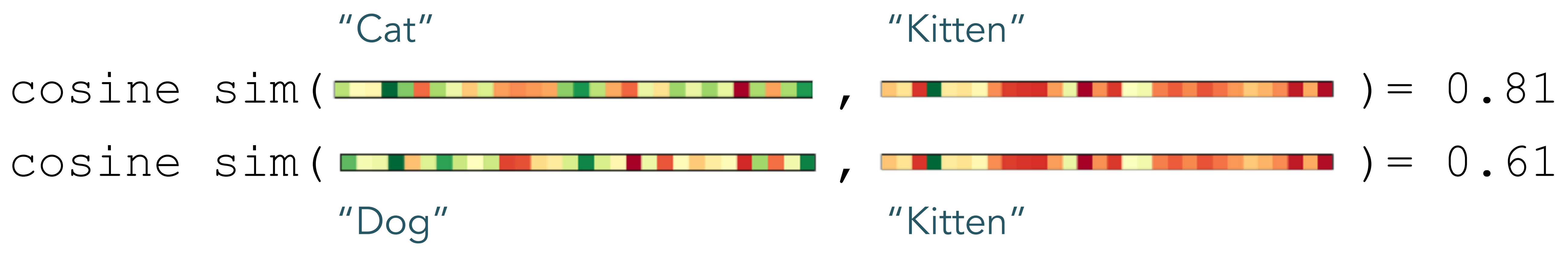
```
▶<div id="item2627" preview_url="https://p.scdn.co/mp3-preview/8f54913108882b
af589270d12840b087ae33adf0" class="genre scanme" scan="true" style="color: #d
382c9; top: 0px; left: 1427px; font-size: 101%" onclick="playx("5K7Dmv4712ml7
tbzvljdTA", "latin tech house", this);" title="e.g. Hector Couto "Salimo"">…
</div>
```

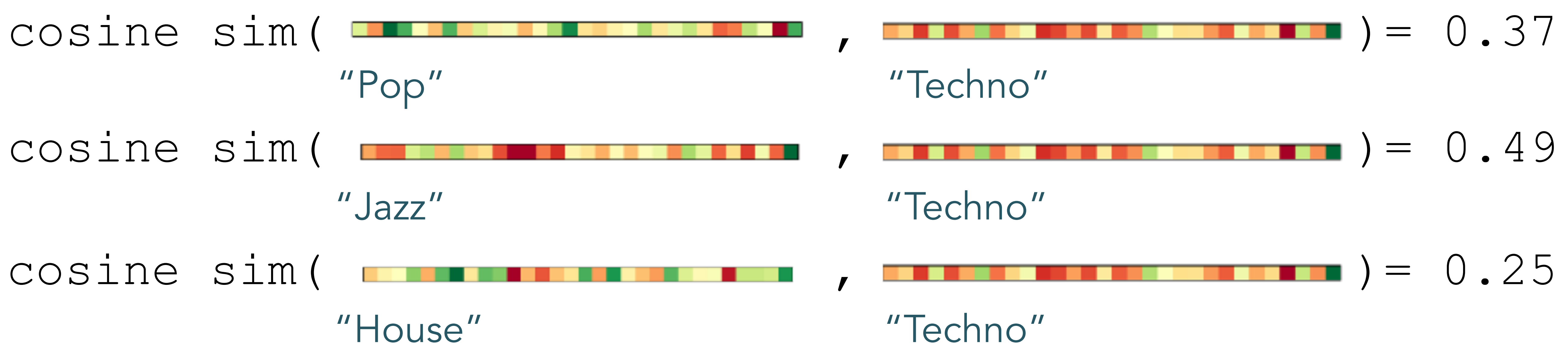
- Song genre embedding equals $(\bar{x}, \bar{y})$, i.e. centroid of all associated genres
- Use Euclidean distance as similarity metric



Genre Similarity
Word2Vec

1. Treat genre as any other word in a natural language, in this case English
2. Apply a pre-trained Word2Vec model to get word embeddings
3. Use cosine similarity as similarity metric





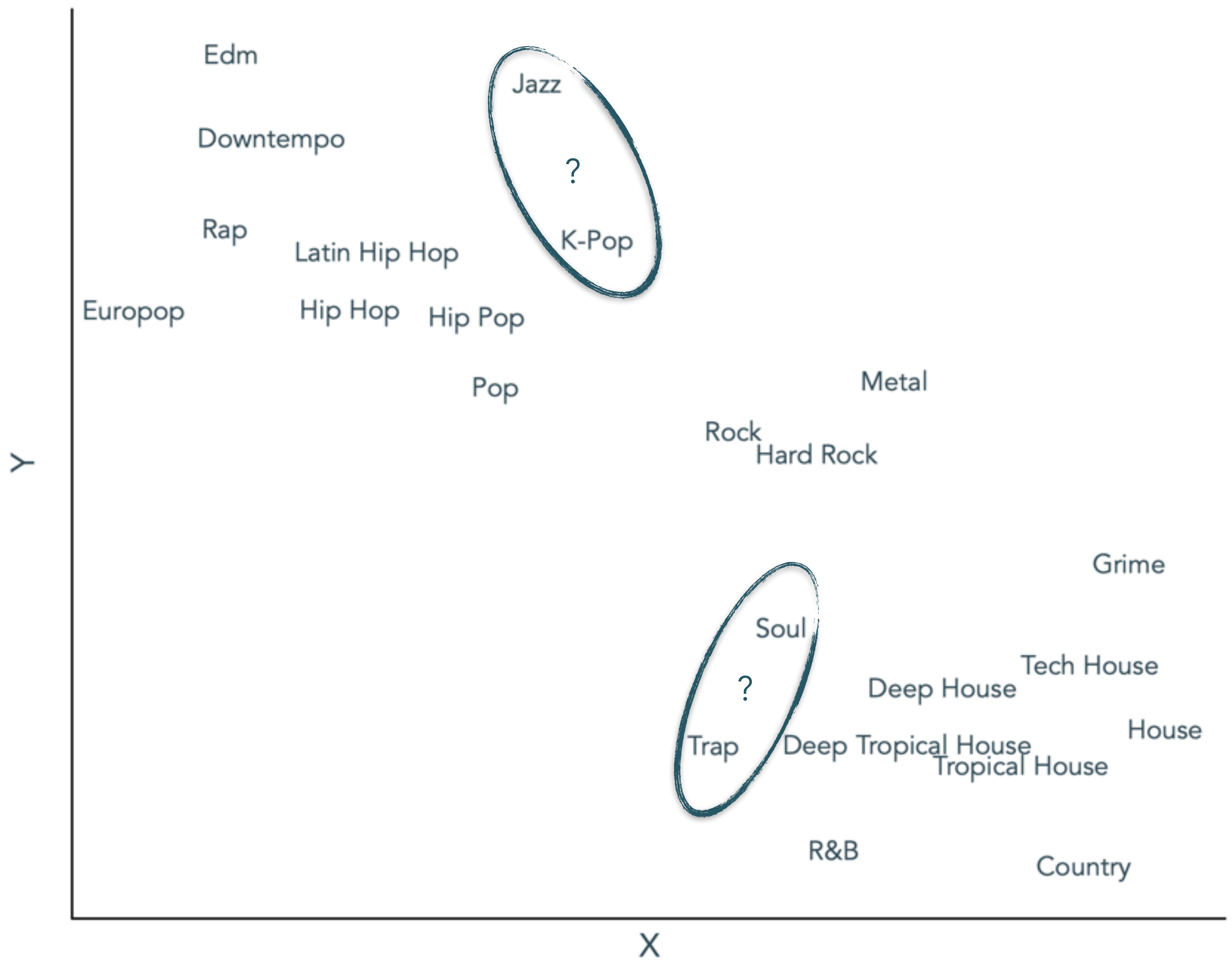
Is this a reasonable approach?

Subjective or not, Jazz is not more similar to Techno than House

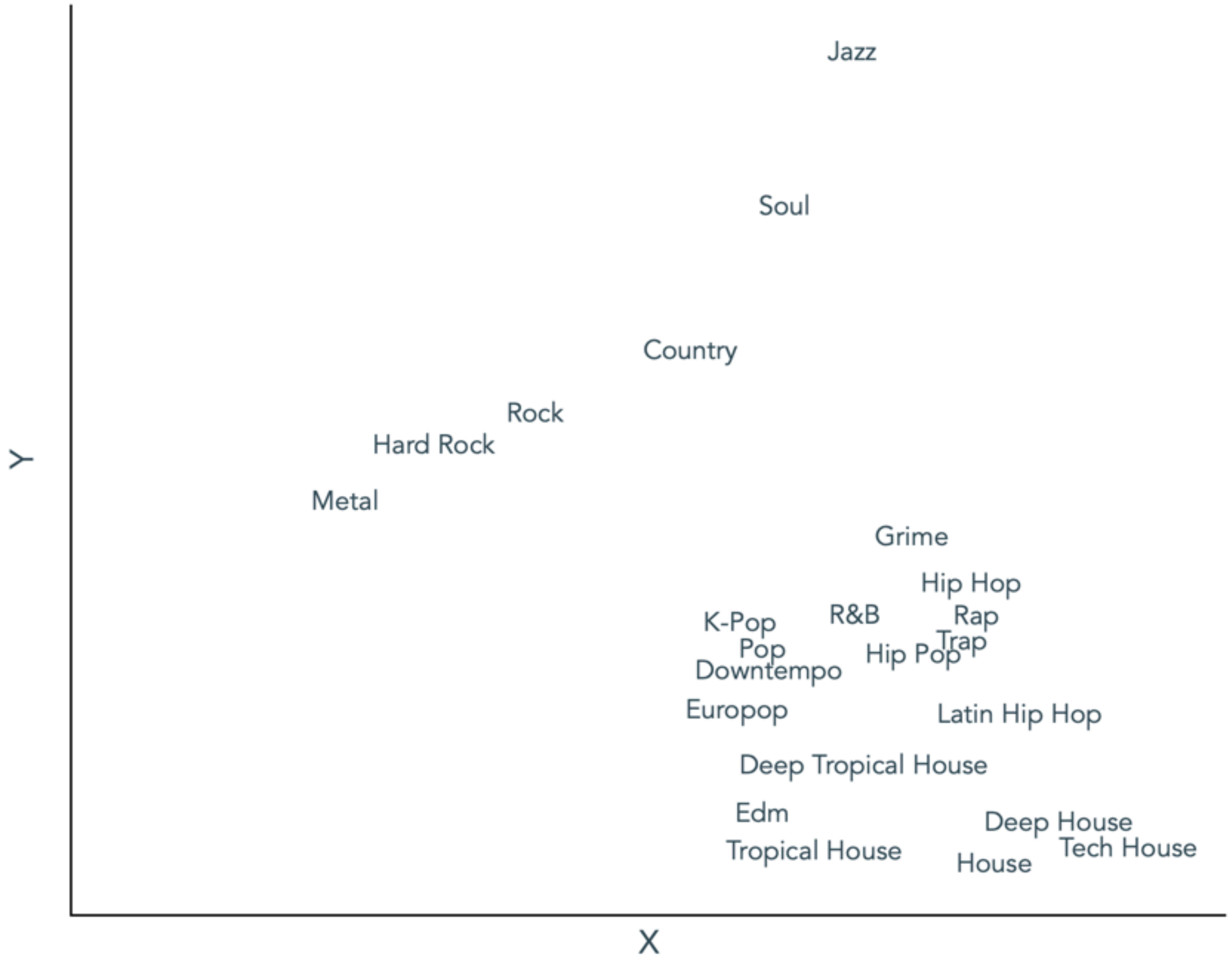
Genre Similarity

Every Noise at Once vs Word2Vec for selected genres

Word2Vec Embedding Space Projected in 2D Using t-SNE



Every Noise At Once Embedding Space



Word2Vec embeddings seem to lead to strange genre similarities

Genre Similarity
Every Noise at Once vs Word2Vec

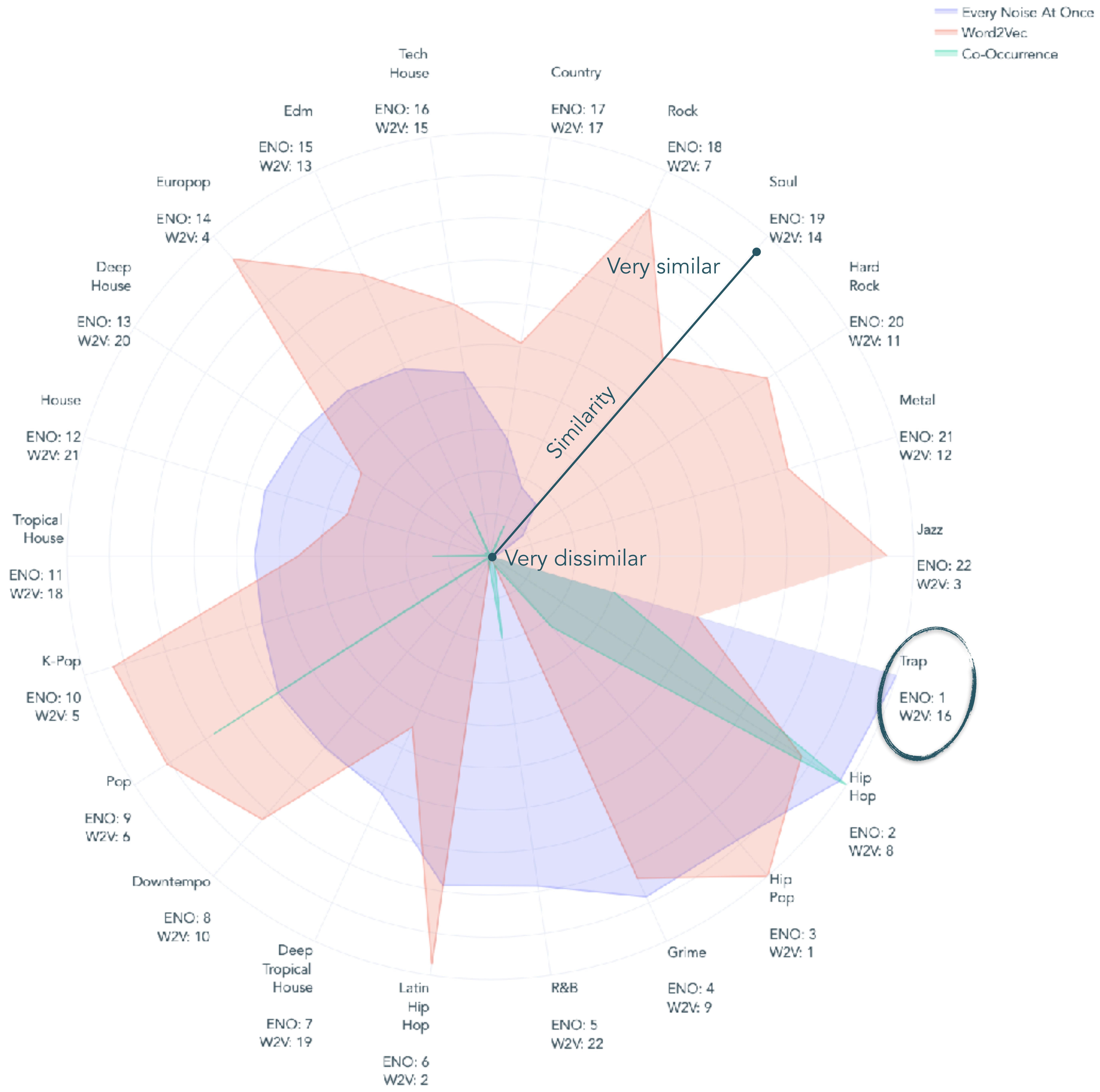
t-SNE introduces distortions

- Difficult to deduce nearest neighbours

Measure genre similarity in original space

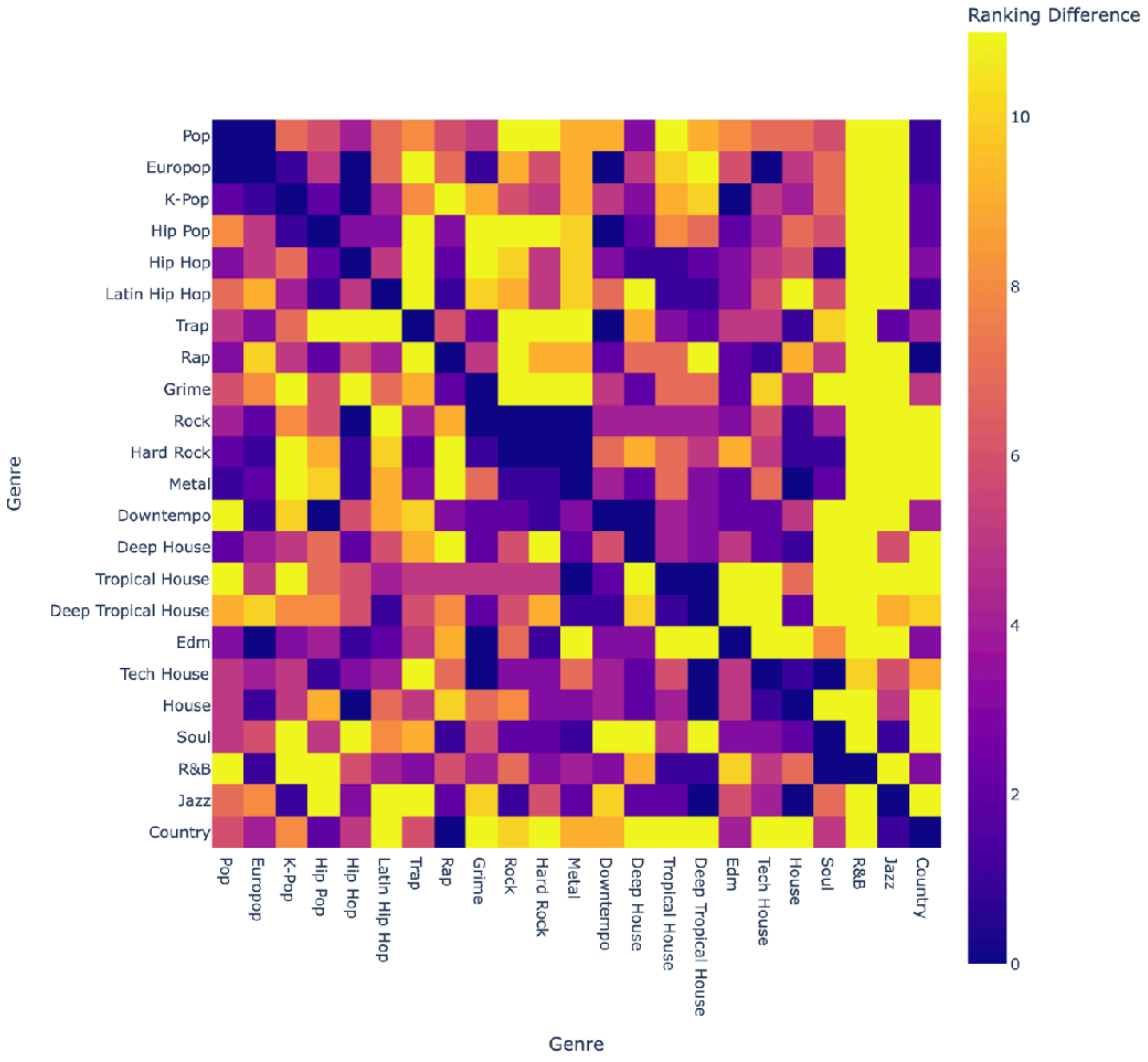
Compare similarity ranks for a given genre

- Choose Rap
- Trap is most similar to Rap according to "Every Noise at Once"
- Word2Vec ranks Trap only 16th in terms of similarity
- The difference in rank is 15, i.e. the two approaches disagree quite considerably



Genre Similarity
Every Noise at Once vs Word2Vec

- Visualise the (dis)agreement between the two methods for all genres at once
- Generally, there is a lot of disagreement (high difference in ranking)
- This is especially the case for R&B and Jazz



Take-away

- Word2vec does not lead to same similarities as Every Noise at Once
- The latter can subjectively be considered as the better embedding approach


YOU BROKE
ME FIRST
Verified Artist
Dash ~~Berlin~~ board
1,356,807 monthly listeners

Recommended Frequencies: A Recommendation System For Playlists

This app allows users to identify songs in their library that may fit well into a selected playlist

Page Settings +

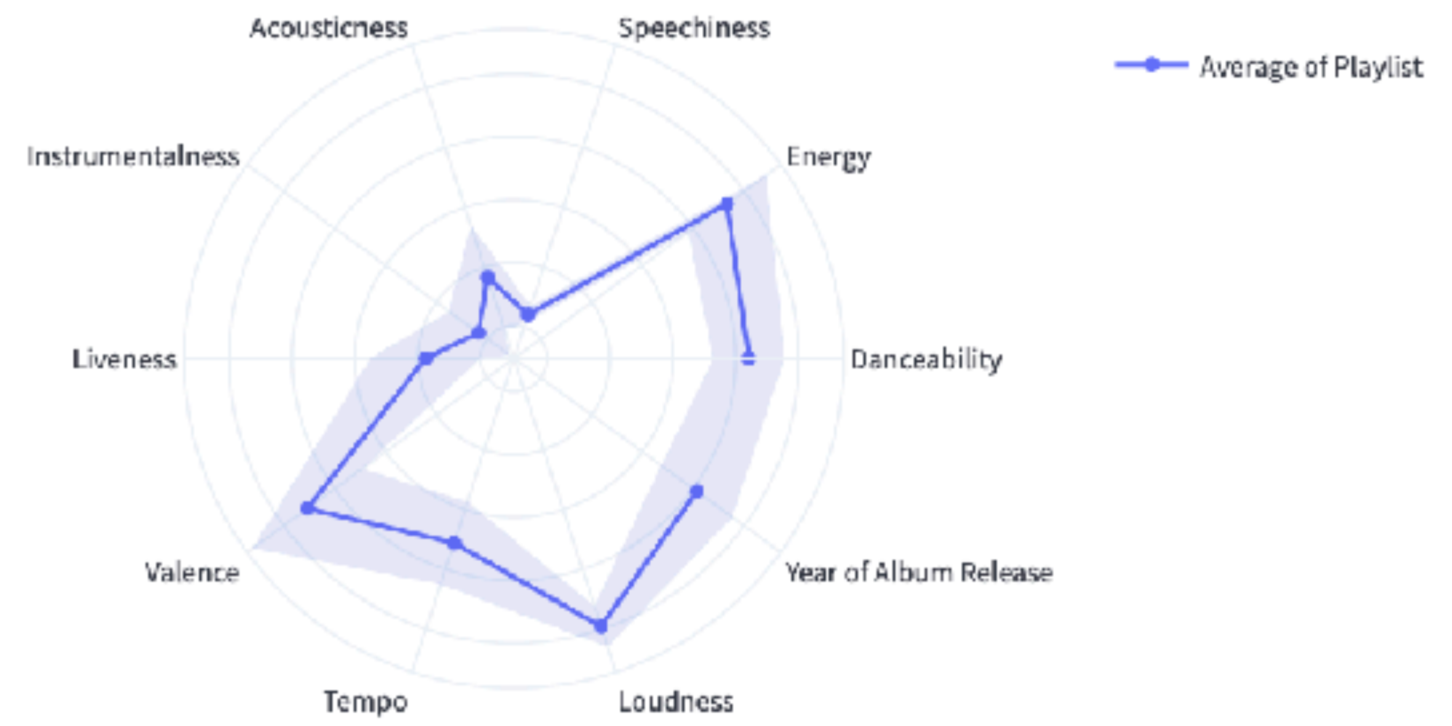
1. Choose a playlist

80's Retro

Examples of songs in playlist «80's Retro»

	SongName	Artist	ID
0	Sledgehammer	Peter Gabriel	029NqmlySn1kOY305AAhxT
1	200度	Sally Yeh	0CYelmjEps63DAuqLV9b6J
2	Stella Stai	Umberto Tozzi	0NUyAEi7WInhF0SJGVavUG
3	Tainted Love	Soft Cell	0cGG2EouYCEEC3xfa0tDFV
4	The Rhythm of the Night	Corona	0ofMkI3jzmGCElAOgOLeo3
5	Just Can't Get Enough	Depeche Mode	0qi4b1l0eT3jpzeNHeFXDT
6	I'm So Excited	The Pointer Sisters	1ot6jEe4w4hYnsOPjd3xKQ
7	Like a Prayer	Madonna	1z3ugFmUKoCzGsI6jdY4Ci
8	Two Tickets to Paradise	Eddie Money	22ClOfLZB9z8He7WgHYAgH
9	Tarzan Boy	Baltimora	273uCXd7NPrlnaiNqtkOrA

Song attributes for playlist «80's Retro»



2. Find similar songs

Similarity Settings

Similarity Features

Numeric Song Attributes + Genre

Genre Similarity Metric

everynoise

Genre Weight

0.30

0.00 1.00

Nr of Suggestions

10

5 20

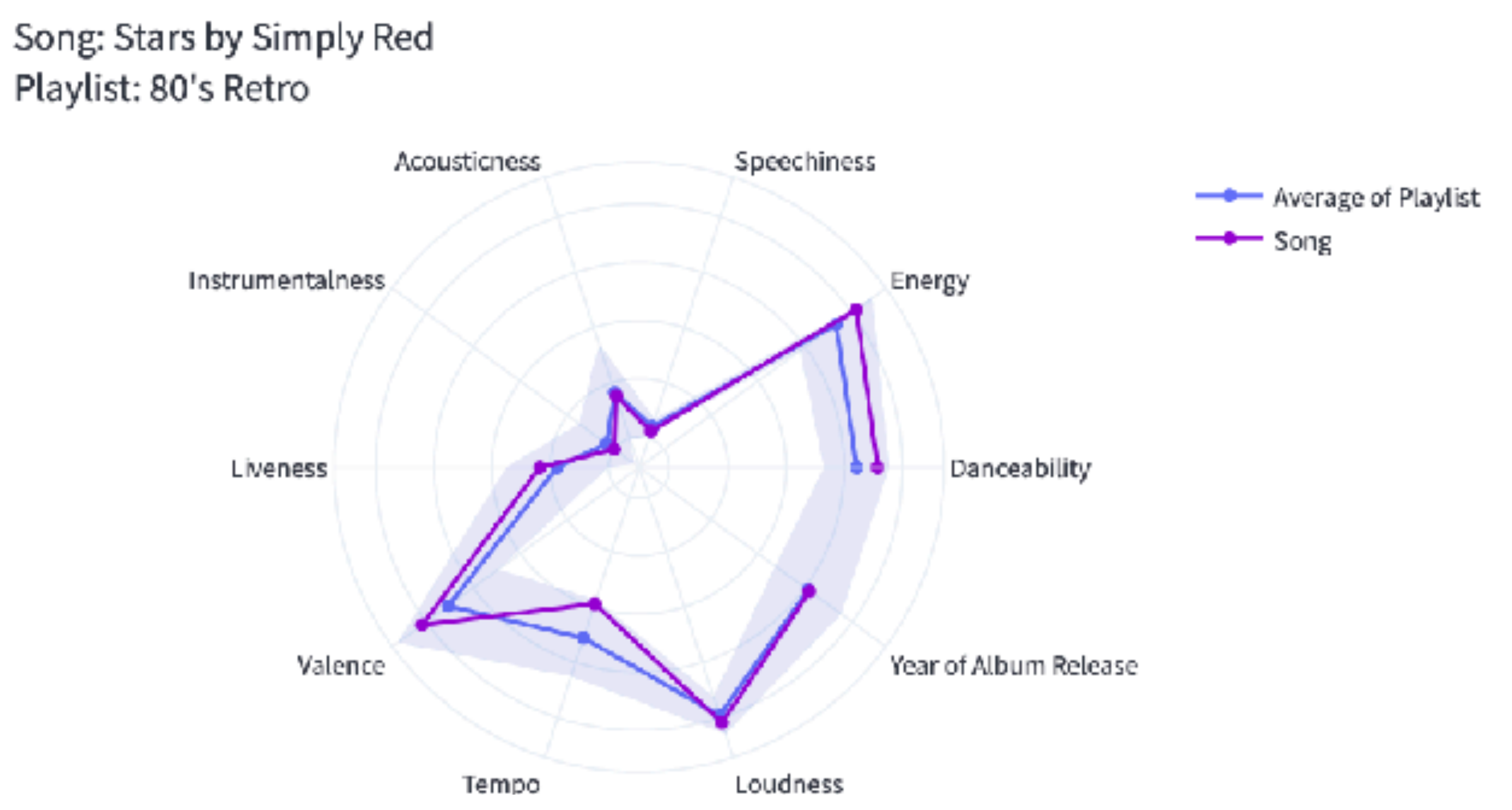
Most similar songs to playlist «80's Retro»

	SongName	Artist	ID
0	Stars	Simply Red	75CgD6l7K4qMzZrn4CbZqz
1	The Best	Tina Turner	6pPWRBubXOBAHnjl5ZlujB
2	How Bizarre	OMC	46q5BtHso0ECuTKeq70ZhW
3	Fairground	Simply Red	3lghqWELEWj9yEPvme7c4W
4	You Get What You Give	New Radicals	1Cwsd5xl8CajJz795oy4XF
5	Save Tonight	Eagle-Eye Cherry	0smyCrJiibi2uwCiq5R1vj
6	Breakfast At Tiffany's	Deep Blue Something	1uzWOoJdADfstQuFtQFTUn
7	If You Love Somebody Set Them Free	Sting	5Xhqe9xu6bKRSqLj1mS1SB
8	Wouldn't It Be Good	Nik Kershaw	00FDHurakzVEiPutdUxXXq
9	In These Shoes?	Kirsty MacColl	6K2exMTxOTgErtQlp5BRMp

Visualise similarity of proposed song

Spotify song ID to visualise

75CgD6l7K4qMzZrn4CbZqz



Listen to proposed song

0:00 0:30

Let's deep dive into some results

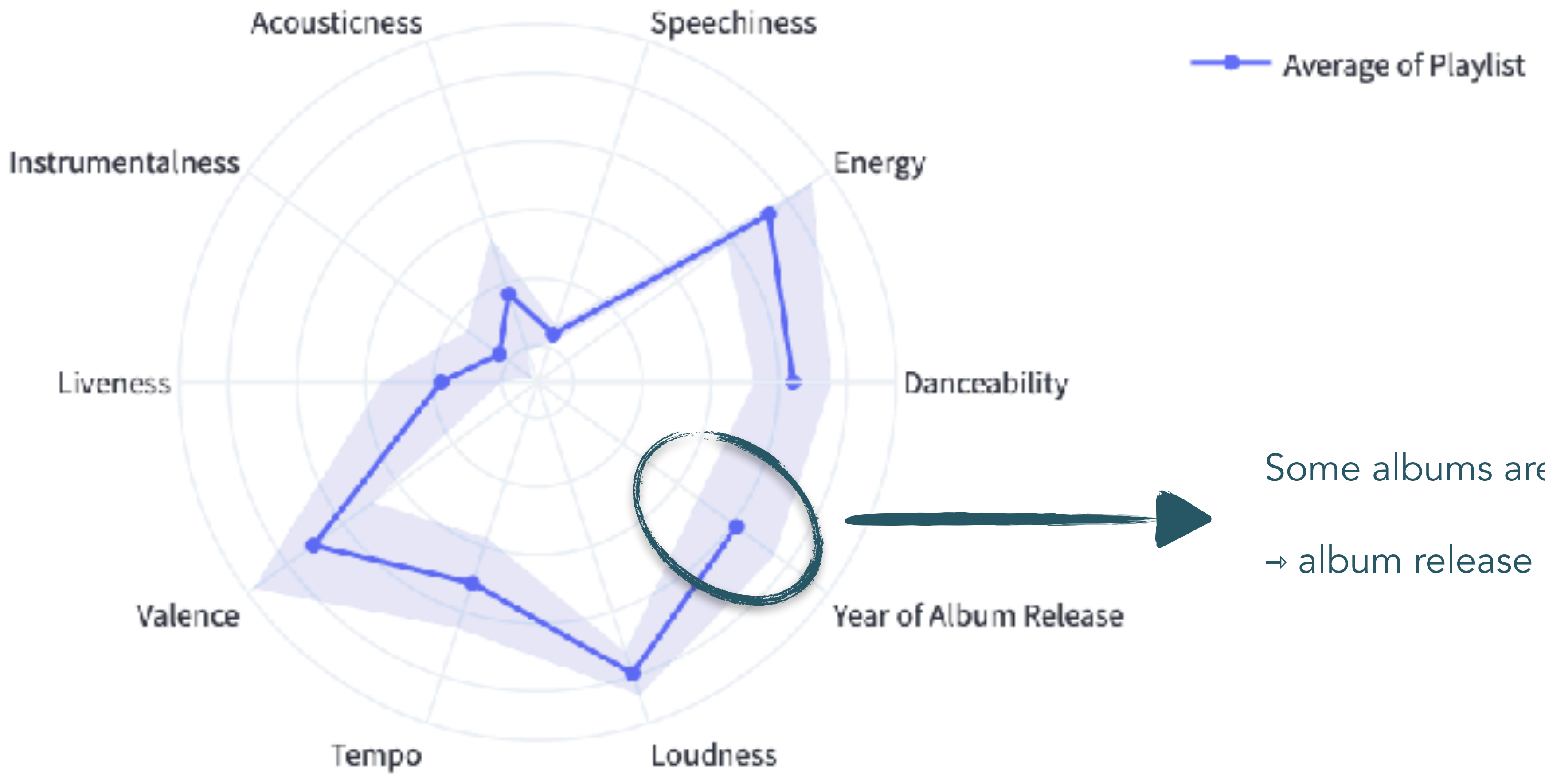
«80's Retro»
for that 80's feeling

Example of songs in the playlist

Sledgehammer	Peter Gabriel
200度	Sally Yeh
Stella Stai	Umberto Tozzi
Tainted Love	Soft Cell
The Rhythm of the Night	Corona
Just Can't Get Enough	Depeche Mode
I'm So Excited	The Pointer Sisters
Like a Prayer	Madonna
Two Tickets to Paradise	Eddie Money
Tarzan Boy	Baltimora

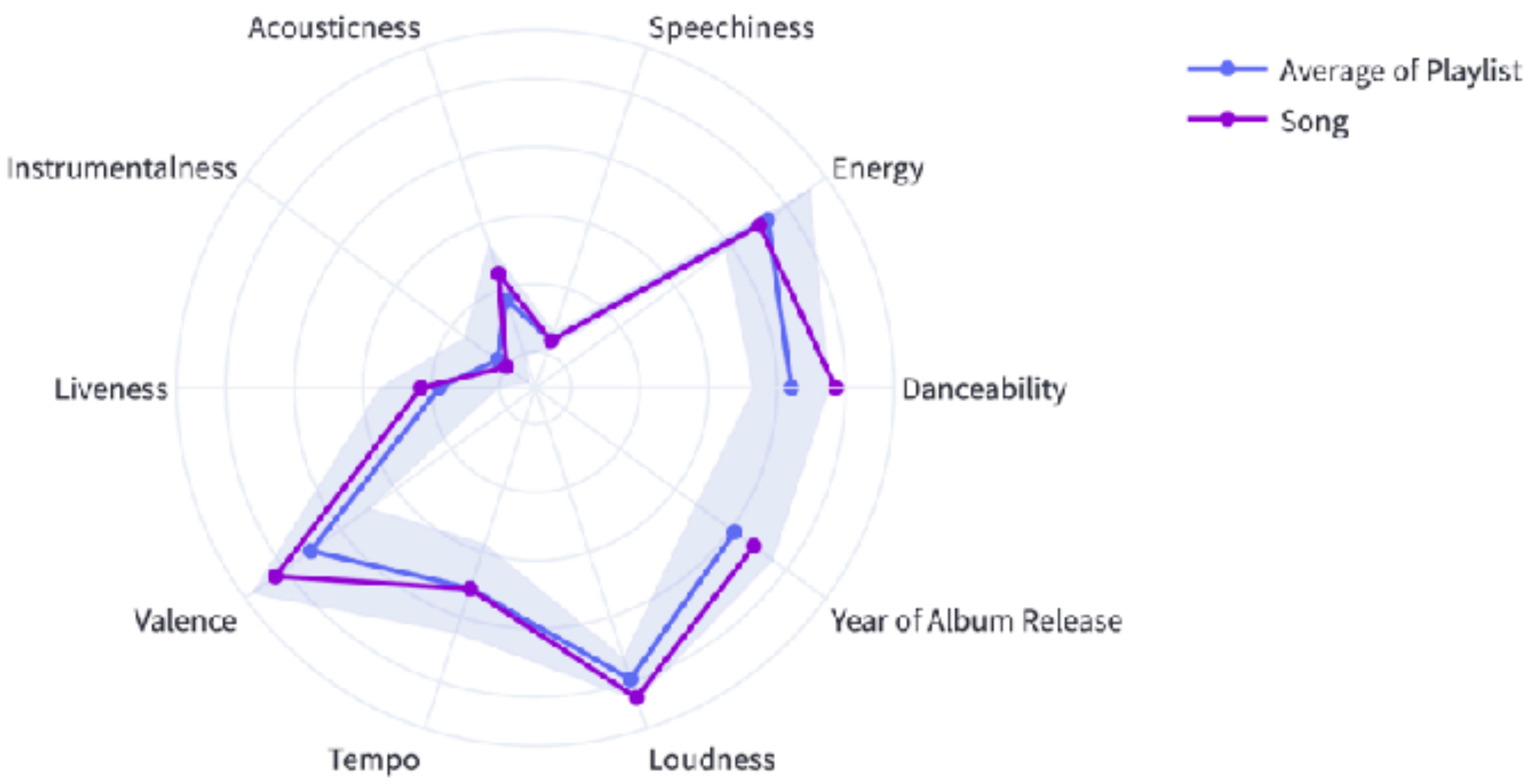
Top recommendations

👍	Stars	Simply Red
👌	The Best	Tina Turner
👎	How Bizarre	OMC
👍	Fairground	Simply Red
👎	You Get What You Give	New Radicals
👎	Save Tonight	Eagle-Eye Cherry
👎	Breakfast At Tiffany's	Deep Blue Something
🤔	If You Love Somebody Set Them Free	Sting
👌	Wouldn't It Be Good	Nik Kershaw
👎	In These Shoes?	Kirsty MacColl



Some albums are remastered

→ album release not in the 80's



Song

Takeaway

The Chainsmokers

- Works decently well for playlists based on acoustic characteristics
- Faces limitations when it comes to playlists based on concepts

Some possible improvements for the

- Include language of songs to filter on
- Address issue with remastered albums by getting oldest release
- Allow specific filters (e.g. acceptable interval of valence)
  - Useful for playlists capturing certain mood

The End