Spotify Recommendation Engine

Tanisha Rajgor - Large-Scale Storage & Retrieval

Model Size: 1192 nodes, 70829 edges, Density: 0.100
Similarity Metric: Cosine Similarity

Summary + Visualization

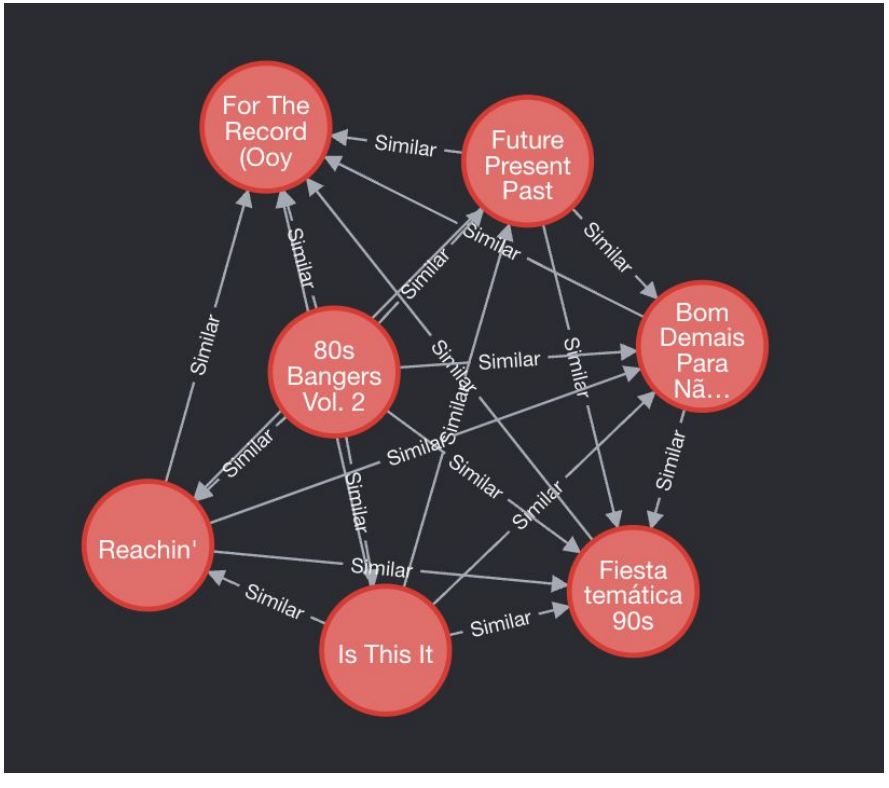
Recommendations (artist, album, track - by top score):

1) Counting Crows, Fiesta temática 90s, Mr. Jones
2) Netto;Isaias Saad, Bom Demais Para Não Crer, Bom Demais Para Não Crer
3) Ooyy;HDBeenDope, For The Record (Ooyy Remix), For The Record (Ooyy Remix)
4) Diana Ross, 80s Bangers Vol. 2, Upside Down
5) Phase II, Reachin', Reachin' - Brotherhood Mix

Approach:

For each of the 114 genres, 10 songs were randomly sampled from each, including songs from “The Strokes” for a total of 1192 songs. Then, for each pair of songs/nodes the cosine similarity scores were calculated based on 10 features (namely danceability, energy, key, loudness, speechiness, acousticness, instrumentalness, liveness, valence, tempo - all quantitative measures). Those that were above the 90th percentile (from all scores) were added as edges. For efficiency, edge repetition and self-pairings were avoided (checking if similarity score = 1).

*Initially, I did try to use ~100 songs for each genre, leading to a 11-12k sample size, however, even with pre-processing (which reduced the number of edge by 84%) it still had an extensively long loading time in Neo4J. Additionally, I first only took the top 25% of edges, however then further reduced it to 10% of edges, in terms of cosine similarity.



The following graph by default shows the album of each track as the face trait. *Is This It* and *Future Present Past* are by “The Strokes” and are just shown for reference for where connections are made.

Cypher Query:

```
MATCH (s: Song)
WHERE s.album_name = 'Is This It'
WITH s
MATCH (s)-[r: Similar]->(similar_song: Song)
RETURN similar_song
WHERE similar_song.artists <> 'The Strokes'
ORDER BY r.weight DESC
LIMIT 5;
```