DATA643 - Final Project

Dan Fanelli

July 15, 2017

Music Recommender with User Filters

This Document

The core work of this project was coded in Scala using the Apache Spark framework + environment. The code is therefore in an Eclipse project, while thie output is in an HTML document, but this document will serve to explain the input, the process, and the output of this final assignment.

The Mission

- Music listeners have different moods, and often want suggestions within a specific genre.
- Sometimes recommender system results can suggest music from a genre that the user was not hoping for (ie - when prediciting for a user intent on a rap song, but r&b is recommended).
- lastfm's data set hetrec2011-lastfm-2k provides a large set of data including user artist "weights", but also with user artist "tags", which provide us a means of filtering by genre.
- Because this is such a large data set, the Apache Spark platform was used
- The Scala programming language was used from an Eclipse IDE to perform the recommendations

The Process

First we query the lastfm tags data set to confirm that this corresponds to music genres that we would like to filter by:

tagValue	count(1)
rock	7503
pop	5418
alternative	5251
electronic	4672
indie	4458
female vocalists	4228
80s	2791
dance	2739
alternative rock	2631
classic rock	2287
british	2092
indie rock	2060
singer-songwriter	1834
hard rock	1789
experimental	1741
metal	1729
ambient	1699
90s	1615
new wave	1595
seen live	1439

only showing top 20 rows

Figure 1:

Tags:

These seem like reasonable music genre's, so we will use these genres in conjunction with the following other data sets to produce our filtered recommendations:

Artists:

```
+---+--------------------+--------------------+--------------------+
| id|                name|                 url|          pictureURL|
+---+--------------------+--------------------+--------------------+
|  1|        MALICE MIZER|http://www.last.f...|http://userserve-...|
|  2|     Diary of Dreams|http://www.last.f...|http://userserve-...|
|  3|   Carpathian Forest|http://www.last.f...|http://userserve-...|
|  4|        Moi dix Mois|http://www.last.f...|http://userserve-...|
|  5|         Bella Morte|http://www.last.f...|http://userserve-...|
|  6|           Moonspell|http://www.last.f...|http://userserve-...|
|  7|      Marilyn Manson|http://www.last.f...|http://userserve-...|
|  8|         DIR EN GREY|http://www.last.f...|http://userserve-...|
|  9|         Combichrist|http://www.last.f...|http://userserve-...|
| 10|             Grendel|http://www.last.f...|http://userserve-...|
| 11|            Agonoize|http://www.last.f...|http://userserve-...|
| 12|            Behemoth|http://www.last.f...|http://userserve-...|
| 13|              Hocico|http://www.last.f...|http://userserve-...|
| 15|        Dimmu Borgir|http://www.last.f...|http://userserve-...|
| 16|London After Midn...|http://www.last.f...|http://userserve-...|
| 17|        Psyclon Nine|http://www.last.f...|http://userserve-...|
| 18|     The Crüxshadows|http://www.last.f...|http://userserve-...|
| 19|          :wumpscut:|http://www.last.f...|http://userserve-...|
| 20|        Limbonic Art|http://www.last.f...|http://userserve-...|
| 21| Artista sconosciuto|http://www.last.f...|http://userserve-...|
+---+--------------------+--------------------+--------------------+
only showing top 20 rows
```

Figure 2:

User__Artists:

User__Tagged__Artists:

The Joins and Filter:

With this data in hand, we do a join of the tables combined with a filter for the specified genre as follows:

```
spark.sql("select uta.userID, user_artists.artistID, artists.name, avg(user_artists.weight) as userArti
```

These 3 fields are used as input to our spark recommender:

userID	artistID	weight
2	51	13883
2	52	11690
2	53	11351
2	54	10300
2	55	8983
2	56	6152
2	57	5955
2	58	4616
2	59	4337
2	60	4147
2	61	3923
2	62	3782
2	63	3735
2	64	3644
2	65	3579
2	66	3312
2	67	3301
2	68	2927
2	69	2720
2	70	2686

only showing top 20 rows

Figure 3:

userID	artistID	tagID	day	month	year
2	52	13	1	4	2009
2	52	15	1	4	2009
2	52	18	1	4	2009
2	52	21	1	4	2009
2	52	41	1	4	2009
2	63	13	1	4	2009
2	63	14	1	4	2009
2	63	23	1	4	2009
2	63	40	1	4	2009
2	73	13	1	4	2009
2	73	14	1	4	2009
2	73	15	1	4	2009
2	73	18	1	4	2009
2	73	20	1	4	2009
2	73	21	1	4	2009
2	73	22	1	4	2009
2	73	26	1	4	2009
2	94	13	1	4	2009
2	94	15	1	4	2009
2	94	20	1	4	2009

only showing top 20 rows

Figure 4:

```
+------+--------+----------------+
|userID|artistID|userArtistWeight|
+------+--------+----------------+
|  1243|    1239|           257.0|
|   507|    5237|           937.0|
|  1672|   14435|           156.0|
|  1832|    1098|          2935.0|
|  1954|     292|         12312.0|
|  1963|     298|           431.0|
|  1073|     212|           667.0|
|   228|    4524|            14.0|
|    12|     344|          5489.0|
|   102|     197|          4658.0|
|  1914|    7266|           571.0|
|   596|    2531|          4360.0|
|  1082|    3057|           401.0|
|  1364|    2873|           199.0|
|  1408|     961|             1.0|
|  1977|     454|           498.0|
|    27|     961|           226.0|
|   159|    1601|          1301.0|
|   916|     301|          2951.0|
|   387|     251|          3070.0|
+------+--------+----------------+
only showing top 20 rows
```

Figure 5:

userID	artistID	userArtistWeight	prediction	rating
336	5803	386.0	-19.414026260375977	386.0
43	1395	548.0	NaN	548.0
1215	3488	2475.0	-89.43989562988281	2475.0
624	5074	290.0	NaN	290.0
236	65	1679.0	-7.34837532043457	1679.0
1832	65	2546.0	18.854896545410156	2546.0
396	6500	1181.0	NaN	1181.0
765	53	356.0	NaN	356.0
1104	6376	145.0	NaN	145.0
936	772	453.0	NaN	453.0
994	8689	83.0	NaN	83.0
1464	81	359.0	NaN	359.0
1885	81	70.0	346.0842590332031	70.0
1215	8977	475.0	NaN	475.0
396	2044	1651.0	NaN	1651.0
1167	12873	57.0	NaN	57.0
1625	15667	16.0	-36.15647888183594	16.0
396	6503	1079.0	NaN	1079.0
1989	18059	78.0	NaN	78.0
1662	1613	285.0	NaN	285.0

only showing top 20 rows

Figure 6:

which yeilds the following numerical analysis:

and the following recommendations for the specified genre, or tag:

The Results

These results are output to the following LARGE SUGGESTIONS HTML DOCUMENT, of which a sample is shown below:

Conclusion

The outputs in the LARGE SUGGESTIONS HTML DOCUMENT definitely represent music by genre, but further, they also seem to show very popular artists per genre. The suggestions matrix generated by spark's recommender can be used to make specific suggestions to an individual user who gives their specific genre/tag constraints.

```
For Tag = [singer-songwriter], We Suggest:
------------------------------------------------
1)Christina Aguilera
2)Shakira
3)Taylor Swift
4)Brandy
5)Britney Spears
6)Placebo
7)Flyleaf
8)Mika
9)Toni Braxton
10)Evanescence
11)TLC
12)U2
13)P!nk
14)Imogen Heap
15)Tweet
16)Keane
17)Alicia Keys
18)Chico Buarque
19)Beyoncé
20)Tom Waits
21)Lily Allen
22)???
23)Lily Allen
24)Joshua Radin
25)Santigold
26)Blake Lewis
27)Mark Lanegan
28)Sufjan Stevens
29)Van der Graaf Generator
30)Mark Owen
31)Regina Spektor
32)Björk
33)Taylor Swift
34)Sean Lennon
35)The Weakerthans
36)Martin L. Gore
37)Maria Mena
38)Jewel
39)Céline Dion
```

Figure 7:

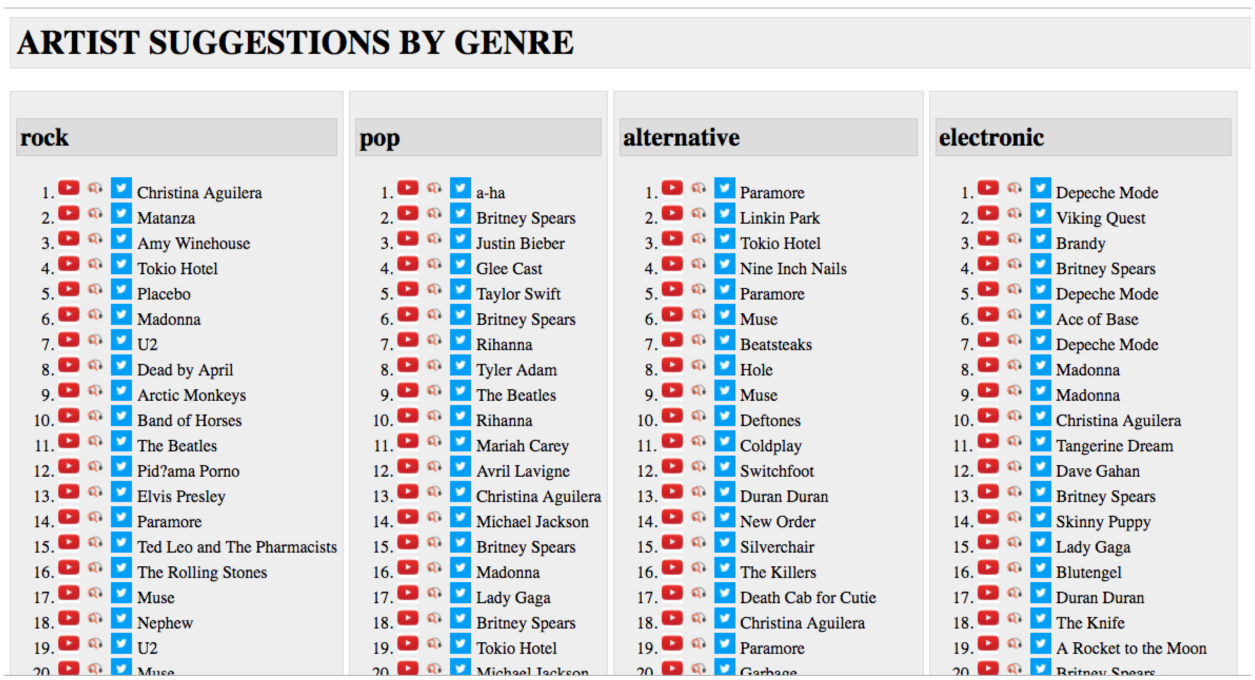
ARTIST SUGGESTIONS BY GENRE
rock
1. Christina Aguilera
2. Matanza
3. Amy Winehouse
4. Tokio Hotel
5. Placebo
6. Madonna
7. U2
8. Dead by April
9. Arctic Monkeys
10. Band of Horses
11. The Beatles
12. Pid?ama Porno
13. Elvis Presley
14. Paramore
15. Ted Leo and The Pharmacists
16. The Rolling Stones
17. Muse
18. Nephew
19. U2
20. Muse
pop
1. a-ha
2. Britney Spears
3. Justin Bieber
4. Glee Cast
5. Taylor Swift
6. Britney Spears
7. Rihanna
8. Tyler Adam
9. The Beatles
10. Rihanna
11. Mariah Carey
12. Avril Lavigne
13. Christina Aguilera
14. Michael Jackson
15. Britney Spears
16. Madonna
17. Lady Gaga
18. Britney Spears
19. Tokio Hotel
20. Michael Jackson
alternative
1. Paramore
2. Linkin Park
3. Tokio Hotel
4. Nine Inch Nails
5. Paramore
6. Muse
7. Beatsteaks
8. Hole
9. Muse
10. Deftones
11. Coldplay
12. Switchfoot
13. Duran Duran
14. New Order
15. Silverchair
16. The Killers
17. Death Cab for Cutie
18. Christina Aguilera
19. Paramore
20. Garbage
electronic
1. Depeche Mode
2. Viking Quest
3. Brandy
4. Britney Spears
5. Depeche Mode
6. Ace of Base
7. Depeche Mode
8. Madonna
9. Madonna
10. Christina Aguilera
11. Tangerine Dream
12. Dave Gahan
13. Britney Spears
14. Skinny Puppy
15. Lady Gaga
16. Blutengel
17. Duran Duran
18. The Knife
19. A Rocket to the Moon
20. Britney Spears

Figure 8: