

lyrics_mapreduce

March 26, 2021

```
[1]: from pyspark.sql import SparkSession
from operator import add

# New API
spark_session = SparkSession\
    .builder\
    .master("spark://192.168.2.111:7077") \
    .appName("lyrics_mapreduce")\
    .config("spark.dynamicAllocation.executorIdleTimeout", "30s")\
    .config("spark.executor.cores", 4)\
    .config("spark.driver.port", 9998)\
    .config("spark.blockManager.port", 10005)\
    .getOrCreate()

# Old API (RDD)
spark_context = spark_session.sparkContext

spark_context.setLogLevel("INFO")

[2]: import pyspark.sql.functions as F

#Get most frequent words for each song ID
lyrics = spark_session.read\
    .option("header", "true")\
    .csv("hdfs://192.168.2.111:9000/user/ubuntu/lyrics_database.csv")\
    .cache()

#Get genre tags for each song ID
lastfm = spark_session.read\
    .json("hdfs://192.168.2.111:9000/user/ubuntu/lastfm/lastfm_test/**/*.  
↪")\
    .repartition(30)\
    .cache()

lyrics.count()
```

[2]: 19045332

```
[ ]:
```

```
[ ]: from pyspark.sql.types import *

#Filter out irrelevant attributes in genre dataset
genre = lastfm.filter(F.size(lastfm["tags"]) > 0 )\
    .select("tags", "track_id")\
    .cache()

genre.count()
```

```
[4]: #Look at data

lyrics.show(3)
genre.show(4)
genre.printSchema()
lyrics.printSchema()
```

```
+---+---+-----+
|word|count|      track_id|
+---+---+-----+
|  i |    6|TRAAAV128F421A322|
| the|    4|TRAAAV128F421A322|
| you|    2|TRAAAV128F421A322|
+---+---+-----+
only showing top 3 rows
```

```
+-----+-----+
|      tags|      track_id|
+-----+-----+
|[[doo wop, 100], ...|TRBHKLA128F930E217|
|[[soul, 100], [mo...|TRDTXAH128F9322744|
|[[Disco, 100], [7...|TRLYCFR128F92DF670|
|[[oldies, 100], [...|TRTIGVQ12903D03BA4|
+-----+-----+
only showing top 4 rows
```

```
root
|-- tags: array (nullable = true)
|   |-- element: array (containsNull = true)
|   |   |-- element: string (containsNull = true)
|-- track_id: string (nullable = true)

root
|-- word: string (nullable = true)
|-- count: string (nullable = true)
|-- track_id: string (nullable = true)
```

[4]: 19045332

```
[5]: #Join both datasets on their ID
paired_songs = lyrics.join(genre, "track_id").cache()
paired_songs.show(4)

paired_songs.count()
```

```
+-----+-----+-----+-----+
|      track_id|word|count|      tags|
+-----+-----+-----+-----+
|TRAADF0128F92E1E91| i|   79|[[dancehall, 100]...|
|TRAADF0128F92E1E91| the|  66|[[dancehall, 100]...|
|TRAADF0128F92E1E91| you|  15|[[dancehall, 100]...|
|TRAADF0128F92E1E91| to|   7|[[dancehall, 100]...|
+-----+-----+-----+-----+

only showing top 4 rows
```

[5]: 1622597

```
[6]: from stop_words import get_stop_words

#Filter stopwords from the frequent words
stopwords = get_stop_words("english")
#ONLY RUN ONCE - For more interesting results
# stopwords.append("just")
# stopwords.append("will")

#Create new DataFrame with a column recasting count to integer
songs_int_count = paired_songs.filter(paired_songs['word'].
    ↳isin(stopwords)==False)\
    .withColumn("wordcount", songs_expanded["count"]\
    ↳cast(IntegerType()))\
    .drop("count")\

#Create new DataFrame that contains one row for each genre tag, the word, and
    ↳the word count in each song
songs_expanded = songs_int_count.select("word",\
    "wordcount",\
    F.explode(paired_songs["tags"]))\
    .withColumnRenamed("col", "genre")\
    .cache()

songs_expanded.show(3)
```

```

+----+-----+-----+
|word|count|          genre|
+----+-----+-----+
|will|   4| [dancehall, 100]|
|will|   4| [raggamuffin, 100]|
|know|   3| [dancehall, 100]|
+----+-----+-----+
only showing top 3 rows

```

```

[7]: import pyspark.sql.types

#Remove second element in tuple to obtain only genre tag
def remove_similarity(genre_tuple):
    genre, _ = genre_tuple

    return genre

tags_function = F.udf(remove_similarity, StringType())

#Cast count to an integer type and remove second element in genre tuple
wordcount_genre = songs_expanded.withColumn("genre",
→tags_function(songs_expanded["genre"]))\
    .cache()

wordcount_genre.show(2)

```

```

+----+-----+-----+
|word|    genre|wordcount|
+----+-----+-----+
|will| dancehall|      4|
|will| raggamuffin|      4|
+----+-----+-----+
only showing top 2 rows

```

```

[8]: # Group elements with common genre and word, and sum their wordcounts
wordcount_genre.groupBy("word", "genre")\
    .agg(F.sum("wordcount"))\
    .sort("sum(wordcount)", ascending=False)\
    .show()

```

```

+----+-----+-----+
|word|    genre|sum(wordcount)|
+----+-----+-----+
|will|    rock|      8996|
|love|    pop|      7432|
|know|    rock|      6546|

```

will	pop	6455
know	pop	5858
love	rock	5674
just	rock	5561
will	alternative	5212
like	rock	5164
oh	pop	4942
now	rock	4826
will	indie	4815
just	pop	4696
go	rock	4691
can	rock	4574
love	Love	4545
time	rock	4541
like	pop	4533
come	rock	4312
will female vocalists		4204

+-----+

only showing top 20 rows

```
[9]: #Group elements by genre and word, to see which pairs are the most frequent
wordcount_genre.groupBy("genre", "word")\
    .count()\
    .sort("count", ascending=False)\
    .show(40)
```

genre	word	count
rock	will	2638
rock	know	2253
rock	just	2171
rock	like	1921
rock	now	1879
rock	time	1850
rock	can	1767
pop	will	1740
pop	know	1713
rock	go	1695
rock	see	1690
rock	come	1627
pop	just	1600
rock	one	1592
rock	love	1566
rock	feel	1526
rock	get	1517
alternative	will	1503

	pop	love	1496
	rock	never	1413
	pop	like	1410
	rock	make	1392
	rock	say	1386
	rock	way	1360
	pop	can	1356
	rock	take	1355
	indie	will	1353
	pop	time	1352
	rock	got	1339
	pop	now	1308
	rock	ca	1297
	pop	go	1290
	pop	see	1253
	alternative	know	1238
	rock	want	1234
	rock	day	1214
	rock	away	1208
	rock	back	1198
	pop	come	1197
	alternative	just	1181

+-----+-----+-----+

only showing top 40 rows

[10]: *# To find most common genres*

```
wordcount_genre.groupBy("genre")\
    .count()\
    .sort("count", ascending=False)\
    .show()
```

+-----+-----+-----+		
	genre	count
+-----+-----+-----+		
	rock	277460
	pop	198881
	alternative	155521
	indie	136971
	female vocalists	121475
	Hip-Hop	118674
	metal	114356
	favorites	112787
	hip hop	98754
	rap	98415
	00s	97088
	Love	95319

```
| alternative rock| 90408|
|      seen live| 81291|
|      beautiful| 75017|
|  male vocalists| 74259|
|      indie rock| 73948|
|      Awesome| 72476|
|singer-songwriter| 72124|
|      dance| 70189|
+-----+-----+
only showing top 20 rows
```

```
[11]: #Print lists of most common words for the top 5 most common genre tags
top_genres = {"rock", "pop", "alternative", "indie", "Hip-Hop"}
genre_top_words = []
```

```
for genre in top_genres:
    #genre_top_words +=
    wordcount_genre.filter(wordcount_genre["genre"] == genre)\
        .groupBy("genre", "word")\
        .agg(F.sum("wordcount"))\
        .sort("sum(wordcount)", ascending=False)\
        .limit(10)\
        .show()
```

```
+-----+-----+-----+
|genre|word|sum(wordcount)|
+-----+-----+-----+
| pop|love|          7432|
| pop|will|          6455|
| pop|know|          5858|
| pop| oh|          4942|
| pop|just|          4696|
| pop|like|          4533|
| pop| can|          4086|
| pop| get|          3790|
| pop| go|          3770|
| pop| now|          3663|
+-----+-----+-----+
```

```
+-----+-----+-----+
|      genre|word|sum(wordcount)|
+-----+-----+-----+
|alternative|will|          5212|
|alternative|know|          3645|
|alternative|like|          3253|
|alternative|just|          3083|
|alternative|love|          2777|
```

alternative now	2648
alternative time	2428
alternative can	2409
alternative go	2383
alternative get	2370

+-----+-----+

genre word sum(wordcount)	
---------------------------	--

+-----+-----+

indie will	4815
indie know	3182
indie just	2712
indie like	2607
indie now	2250
indie love	2203
indie can	2043
indie oh	2037
indie time	1981
indie go	1980

+-----+-----+

genre word sum(wordcount)	
---------------------------	--

+-----+-----+

rock will	8996
rock know	6546
rock love	5674
rock just	5561
rock like	5164
rock now	4826
rock go	4691
rock can	4574
rock time	4541
rock come	4312

+-----+-----+

genre word sum(wordcount)	
---------------------------	--

+-----+-----+

Hip-Hop like	3381
Hip-Hop get	2774
Hip-Hop la	2027
Hip-Hop got	1991
Hip-Hop know	1953
Hip-Hop de	1826
Hip-Hop just	1709
Hip-Hop will	1669

Hip-Hop	one	1478
Hip-Hop	now	1418
+-----+-----+-----+		

```
[3]: spark_context.stop()
```

```
[ ]: import pandas as pd
import matplotlib.pyplot as plt
```

```
#print(genre_top_words)
```

```
#df = pd.DataFrame({'word': ['word1', 'word2'], 'count': [12898, 4861]})
#df.plot.bar(x='word', y='count', rot=0)
```

```
[3]: df = spark_session.read.format('jdbc').options(url='hdfs://master:9000/user/
↳ubuntu/jdbc:sqlite:mxm_dataset.db',dbtable='lyrics',driver='org.sqlite.
↳JDBC').load()

df.show(2)
```

```
-----
IllegalArgumentExcep          Traceback (most recent call last)
<ipython-input-3-3df2728b08f6> in <module>
----> 1 df = spark_session.read.format('jdbc').options(url='hdfs://master:9000/
↳user/ubuntu/jdbc:sqlite:mxm_dataset.db',dbtable='lyrics',driver='org.sqlite.
↳JDBC').load()
      2
      3 df.show(2)

~/local/lib/python3.8/site-packages/pyspark/sql/readwriter.py in load(self,
↳path, format, schema, **options)
    208         return self._df(self._jreader.load(self._spark._sc._jvm.
↳PythonUtils.toSeq(path)))
    209     else:
--> 210         return self._df(self._jreader.load())
    211
    212     def json(self, path, schema=None, primitivesAsString=None,
↳prefersDecimal=None,

~/local/lib/python3.8/site-packages/py4j/java_gateway.py in __call__(self,
↳*args)
    1302
    1303         answer = self.gateway_client.send_command(command)
-> 1304         return_value = get_return_value(
    1305             answer, self.gateway_client, self.target_id, self.name)
    1306
```

```
~/local/lib/python3.8/site-packages/pyspark/sql/utils.py in deco(*a, **kw)
    115             # Hide where the exception came from that shows a
↳non-Pythonic
    116             # JVM exception message.
--> 117             raise converted from None
    118         else:
    119             raise

IllegalArgumentExce␣ption: requirement failed: The driver could not open a JDBC␣
↳connection. Check the URL: hdfs://master:9000/user/ubuntu/jdbc:sqlite:
↳mxm_dataset.db
```

[]: