Movie Recommendation with MLib

San Francisco Bay University
Dipali Gajera

Contents

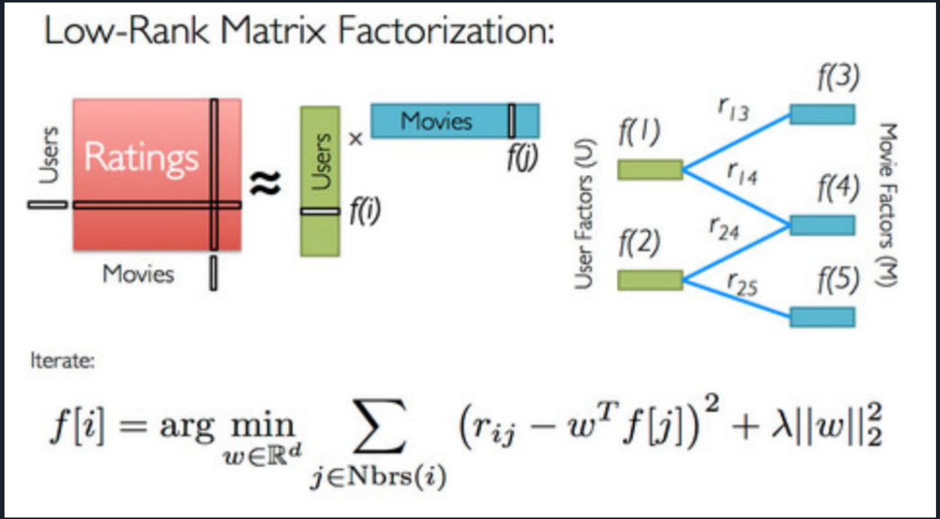
INTRODUCTION

- Machine learning is carried out through Apache Spark's Spark MLlib. The algorithms and tools of MLlib are widely used.
- The original RDD-based API is available in spark.mllib. It's in maintenance mode right now.
- For the purpose of creating ML pipelines, spark.ml offers higher level API built on top of DataFrames. As of right now, Spark's main Machine Learning API is spark.ml.

COLLABORATIVE FILTERING

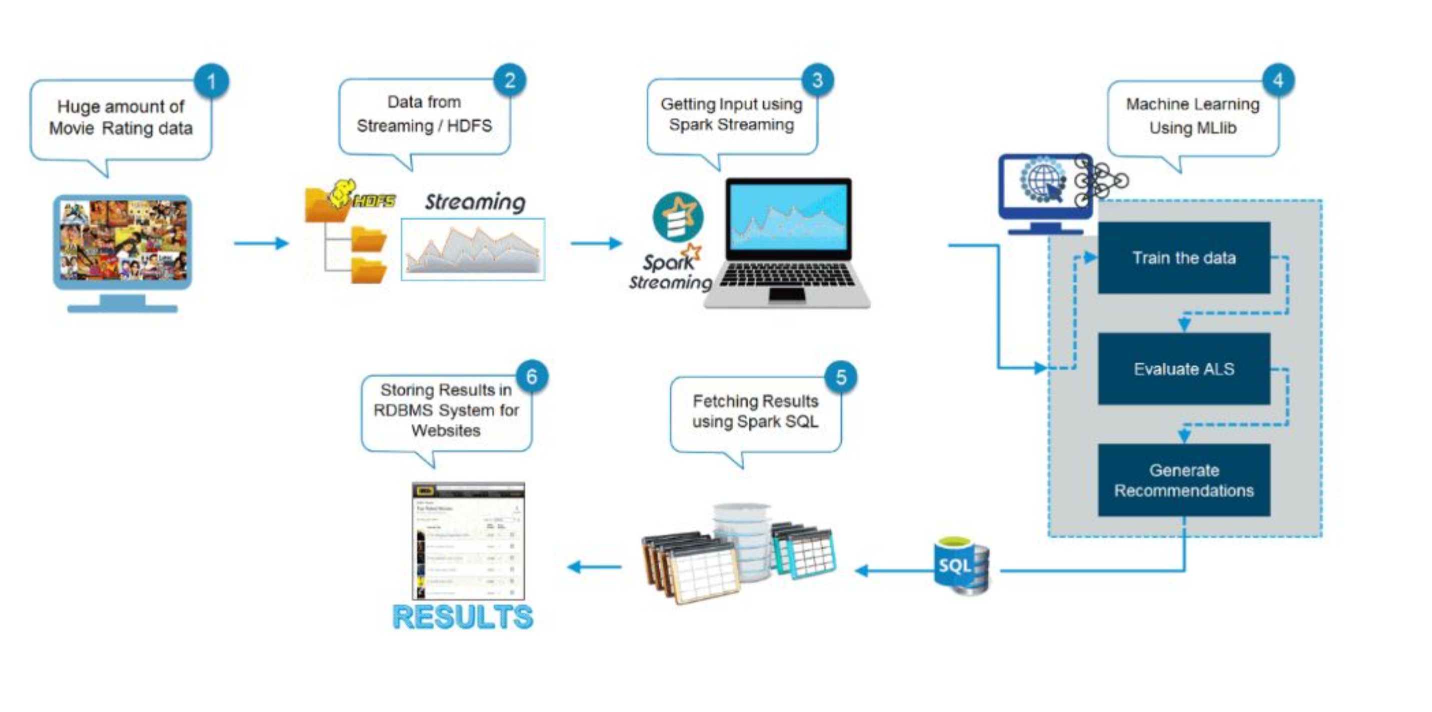
- ❖ Recommender systems frequently use collaborative filtering.
- ❖ In our scenario, the user-movie rating matrix, these strategies seek to complete the gaps in a user-item association matrix.
- ❖ A limited number of latent characteristics that can be used to forecast missing entries are utilized to describe persons and products in the model-based collaborative filtering that MLlib currently offers.

- We will use MLlib to make personalized movie recommendations tailored for you.
- Using data gathered by MovieLens from 72,000 individuals who rated 10,000 films, we will use 10 million ratings.
- The HDFS on your cluster already has this dataset loaded.



DESIGN

- To quickly process vast amounts of data, the best Big Data technology is required. As a result, Apache Spark is the ideal tool for putting our movie recommendation system into practice.



- If user "A" enjoys "Avtar," "Power Rangers," and "Captain America,".
- User "B" enjoys "Spiderman - No Way Home," "Thor," and "Hulk".
- Then we can infer that they share interests in super-hero films.
- Therefore, there is a strong likelihood that user "A" would enjoy "Thor" and user "B" would enjoy "The Avenger."

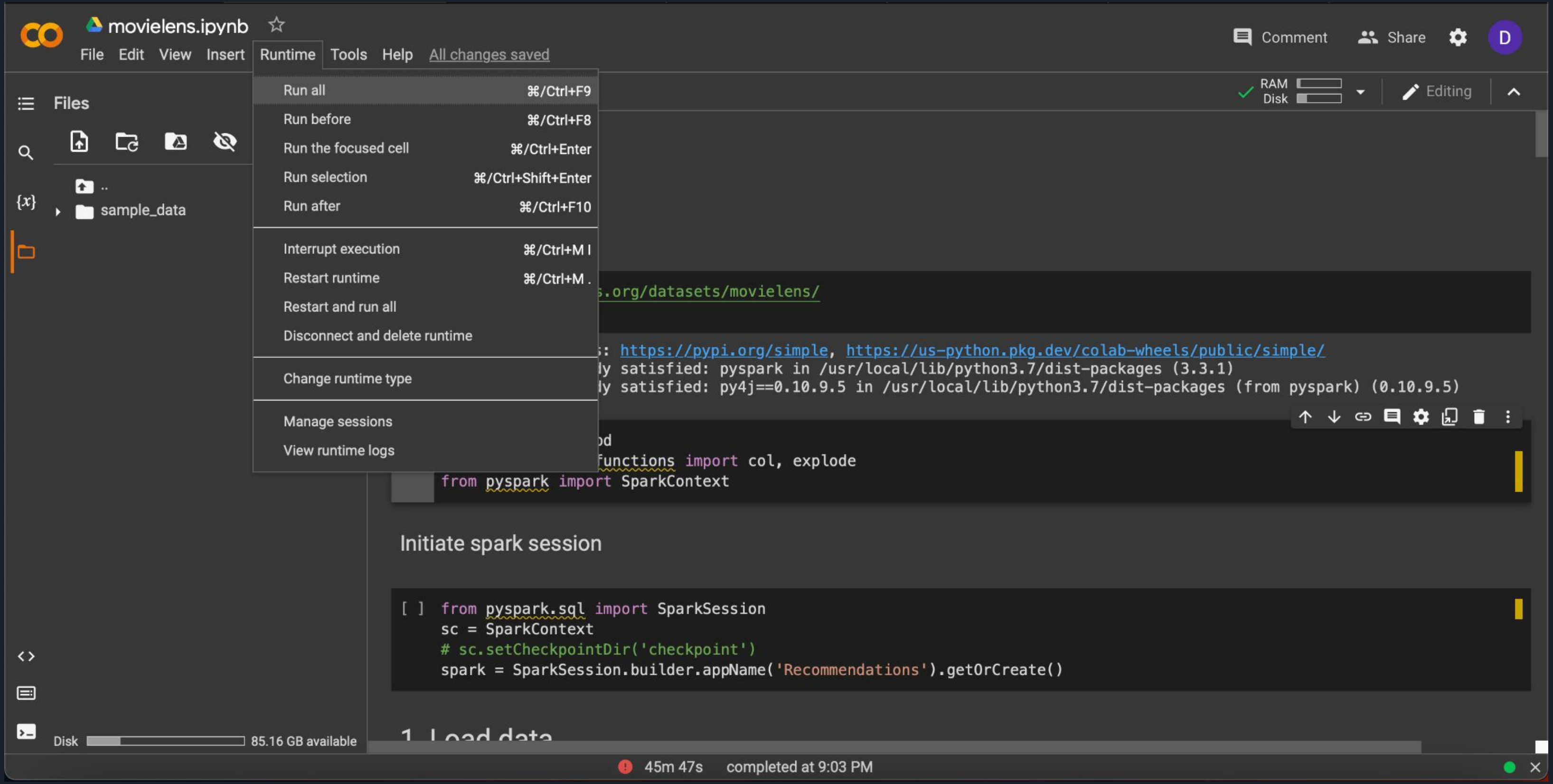
IMPLEMENTATION

- Using Google Collab:

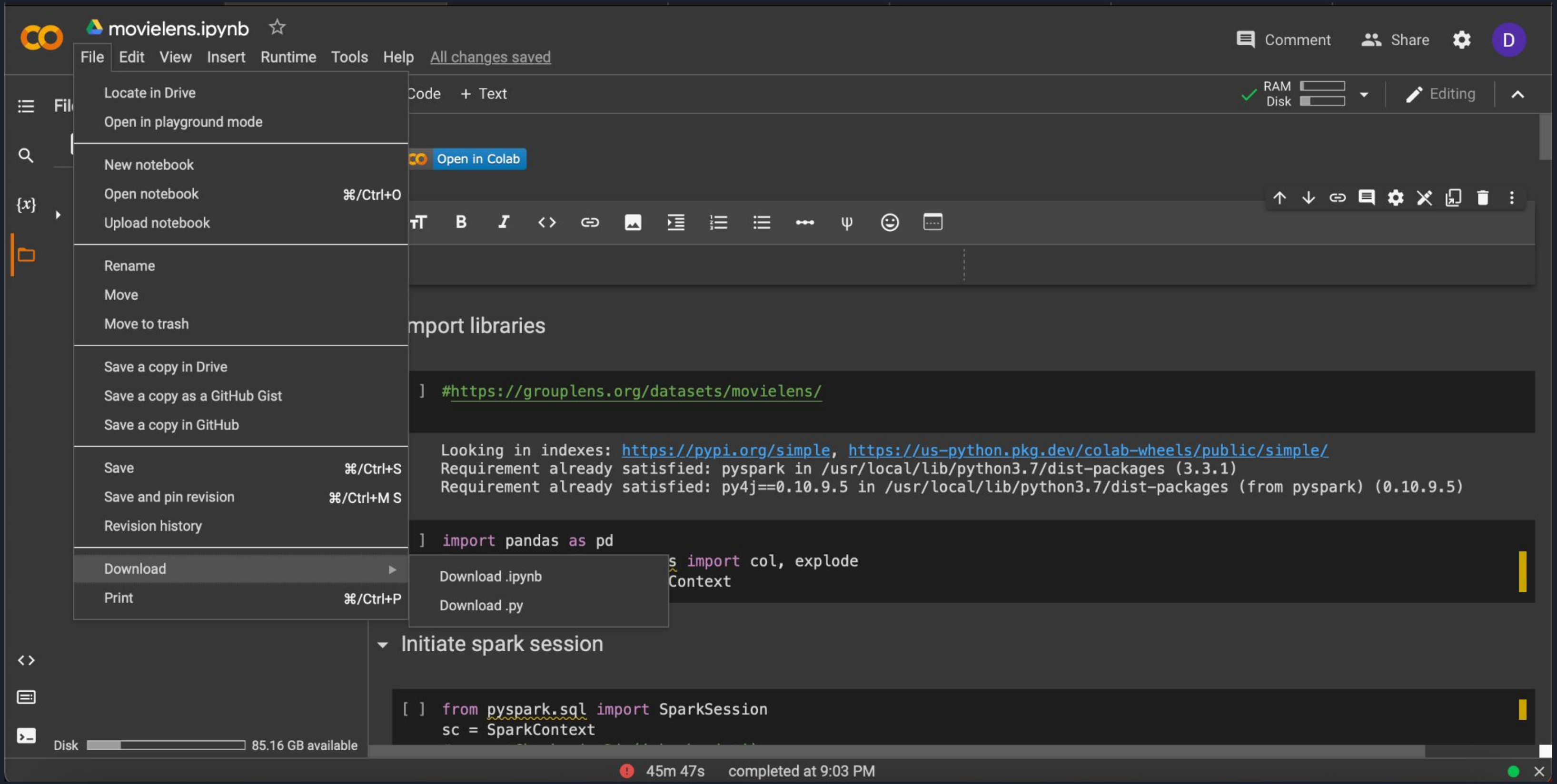
  https://colab.research.google.com/drive/1EWS_faK3UUnNSQNODTg4w3YJ31uRY-7R#scrollTo=WBR2Wjia5b5J

- Upload Files and run all files. Such as,
  - Ipynb
  - Movie.csv
  - Rating.csv
  - tag.csv

Google Collab Platform



❖ Download .py file



Google Cloud Platform

- Go to your google cloud Dataporc cluster

- Publish the files movies.csv, ratings.csv and movielens.py
- Create the HDFS Directory;
  - hdfs dfs -mkdir hdfs:/movielens

- Movies.csv and Ratings.csv should be copied into HDFS Directory:
- movies.csv hdfs:/movielens hdfs dfs -put
- hdfs dfs -put ratings.csv movielens
- Run movielens.py under spark

```
dipaligajera2727@clustermapreduce-m:~$ spark-submit movielens.py
22/11/21 01:27:26 INFO org.apache.spark.SparkEnv: Registering MapOutputTracker
22/11/21 01:27:26 INFO org.apache.spark.SparkEnv: Registering BlockManagerMaster
22/11/21 01:27:26 INFO org.apache.spark.SparkEnv: Registering BlockManagerMasterHeartb
22/11/21 01:27:26 INFO org.apache.spark.SparkEnv: Registering OutputCommitCoordinator
22/11/21 01:27:26 INFO org.sparkproject.jetty.util.log: Logging initialized @4138ms to
22/11/21 01:27:26 INFO org.sparkproject.jetty.server.Server: jetty-9.4.40.v20210413; b
b74; jvm 1.8.0_352-b08
22/11/21 01:27:26 INFO org.sparkproject.jetty.server.Server: Started @4277ms
22/11/21 01:27:26 INFO org.sparkproject.jetty.server.AbstractConnector: Started Server
22/11/21 01:27:27 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceMa
22/11/21 01:27:27 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to Applicati
22/11/21 01:27:29 INFO org.apache.hadoop.conf.Configuration: resource-types.xml not fo
22/11/21 01:27:29 INFO org.apache.hadoop.yarn.util.resource.ResourceUtils: Unable to f
22/11/21 01:27:29 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitte
22/11/21 01:27:30 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceMa
```

TEST

Google Collab Platform

```
# Generate n Recommendations for all users
nrecommendations = best_model.recommendForAllUsers(10)
nrecommendations.limit(10).show()
```

```
+------+--------------------+
|userId|     recommendations|
+------+--------------------+
|     1|[{3379, 5.729532}...|
|     3|[{5746, 4.8612}, ...|
|     5|[{3379, 4.529168}...|
|     6|[{42730, 4.758306...|
|     9|[{3379, 4.921908}...|
|    12|[{42730, 5.673235...|
|    13|[{3379, 5.06624},...|
|    15|[{3379, 4.448795}...|
|    16|[{3379, 4.6072893...|
|    17|[{3379, 5.1776514...|
+------+--------------------+
```

[21]

```
nrecommendations.join(movies, on='movieId').filter('userId = 100').show()
```

```
+-------+------+---------+--------------------+--------------------+
|movieId|userId|   rating|               title|              genres|
+-------+------+---------+--------------------+--------------------+
|  67618|   100| 5.108419|Strictly Sexual (...|Comedy|Drama|Romance|
|  33649|   100|5.0640206|  Saving Face (2004)|Comedy|Drama|Romance|
|   3379|   100|5.0374746| On the Beach (1959)|               Drama|
|  74282|   100|4.9346504|Anne of Green Gab...|Children|Drama|Ro...|
|  42730|   100|4.9183536|   Glory Road (2006)|               Drama|
|  93008|   100| 4.881788|Very Potter Seque...|      Comedy|Musical|
|  25906|   100| 4.881788|Mr. Skeffington (...|       Drama|Romance|
|  77846|   100| 4.881788| 12 Angry Men (1997)|         Crime|Drama|
|   7121|   100|4.8749967|   Adam's Rib (1949)|      Comedy|Romance|
| 171495|   100| 4.874456|              Cosmos|  (no genres listed)|
+-------+------+---------+--------------------+--------------------+
```

```
ratings.join(movies, on='movieId').filter('userId = 100').sort('rating', ascending=False).limit(10).show()
```

```
+-------+------+------+--------------------+--------------------+
|movieId|userId|rating|               title|              genres|
+-------+------+------+--------------------+--------------------+
|   1101|   100|   5.0|      Top Gun (1986)|      Action|Romance|
|   1958|   100|   5.0|Terms of Endearme...|        Comedy|Drama|
|   2423|   100|   5.0|Christmas Vacatio...|              Comedy|
|   4041|   100|   5.0|Officer and a Gen...|       Drama|Romance|
|   5620|   100|   5.0|Sweet Home Alabam...|      Comedy|Romance|
|    368|   100|   4.5|     Maverick (1994)|Adventure|Comedy|...|
|    934|   100|   4.5|Father of the Bri...|              Comedy|
|    539|   100|   4.5|Sleepless in Seat...|Comedy|Drama|Romance|
|     16|   100|   4.5|       Casino (1995)|         Crime|Drama|
|    553|   100|   4.5|    Tombstone (1993)|Action|Drama|Western|
+-------+------+------+--------------------+--------------------+
```

Google Cloud Platform

```
+------+--------------------+
|userId|     recommendations|
+------+--------------------+
|    91|[{3379, 4.9286127...|
|   601|[{3379, 5.447586}...|
|   111|[{128914, 4.82704...|
|   291|[{87234, 5.526545...|
|   581|[{3379, 5.1550307...|
|     1|[{3379, 5.7632384...|
|   223|[{33649, 4.224575...|
|   333|[{3567, 4.7874923...|
|   493|[{876, 4.8167825}...|
|    93|[{3379, 5.7609735...|
+------+--------------------+
```

userId	movieId	rating
471	3379	4.822564
471	8477	4.6659493
471	33649	4.5504856
471	102217	4.5333
471	92494	4.5333
471	33779	4.5333
471	171495	4.527984
471	7096	4.4821672
471	84273	4.4345856
471	117531	4.4345856

movieId	userId	rating	title	genres
67618	100	5.1201425	Strictly Sexual (...	Comedy|Drama|Romance
3379	100	5.064743	On the Beach (1959)	Drama
42730	100	5.042285	Glory Road (2006)	Drama
33649	100	5.021657	Saving Face (2004)	Comedy|Drama|Romance
117531	100	4.9267745	Watermark (2014)	Documentary
7071	100	4.9267745	Woman Under the I...	Drama
184245	100	4.9267745	De platte jungle ...	Documentary
26073	100	4.9267745	Human Condition I...	Drama|War
179135	100	4.9267745	Blue Planet II (2...	Documentary
84273	100	4.9267745	Zeitgeist: Moving...	Documentary

movieId	userId	rating	title	genres
1101	100	5.0	Top Gun (1986)	Action|Romance
1958	100	5.0	Terms of Endearme...	Comedy|Drama
2423	100	5.0	Christmas Vacatio...	Comedy
4041	100	5.0	Officer and a Gen...	Drama|Romance
5620	100	5.0	Sweet Home Alabam...	Comedy|Romance
368	100	4.5	Maverick (1994)	Adventure|Comedy|...
934	100	4.5	Father of the Bri...	Comedy
539	100	4.5	Sleepless in Seat...	Comedy|Drama|Romance
16	100	4.5	Casino (1995)	Crime|Drama
553	100	4.5	Tombstone (1993)	Action|Drama|Western

CONCLUSION

- ❖ We first reviewed the movie lens dataset after gaining theoretical understanding of recommendation engines.
- ❖ Then, after learning how to use MLlib to build collaborative filtering, we divided the dataset into training and testing sets for the transformation tasks.
- ❖ The technique for suggesting movies has enormous potential. For certain people, movie recommendations have been fairly accurate, and movie titles have been successfully clustered based on their plot summaries.

REFERENCE

- https://medium.com/edureka/spark-mllib-e87546ac268
- https://www.linkedin.com/pulse/hands-on-movie-recommendation-spark-mlib-diego-marinho-de-oliveira?trk=prof-post
- https://medium.com/edureka/spark-mllib-e87546ac268
- https://hc.labnet.sfbu.edu/~henry/npu/classes/mllib/collaborative_filtering/PySpark_Recommender_System_with_ALS.pdf

THANK YOU