ex55

August 18, 2022

[ ]:
```
from pyspark import SparkContext, SparkConf
from pyspark.sql import SparkSession
from graphframes import GraphFrame

conf = SparkConf().setAppName("ex55")
sc = SparkContext(conf=conf)
ssql = SparkSession.builder.getOrCreate()
```

[2]:
```
edgesPath = "data/Ex55/data/edges.csv"
vertexesPath = "data/Ex55/data/vertexes.csv"
outputPath = "out55/"
```

[3]:
```
eDF = ssql.read.load(
    edgesPath,
    format="csv",
    header=True,
    inferSchema=True
)

vDF = ssql.read.load(
    vertexesPath,
    format="csv",
    header=True,
    inferSchema=True
)
```

[4]:
```
eDF.printSchema(), vDF.printSchema()
```

```
root
 |-- src: string (nullable = true)
 |-- dst: string (nullable = true)
 |-- linktype: string (nullable = true)

root
 |-- id: string (nullable = true)
 |-- entityName: string (nullable = true)
 |-- name: string (nullable = true)
```

[4]: (None, None)

[5]:
```
eDF.show(), vDF.show()
```

```
+---+---+----------+
|src|dst|  linktype|
+---+---+----------+
| V1| V2|      like|
| V1| V3|    follow|
| V1| V4|    follow|
| V3| V2|    follow|
| V3| V4|    follow|
| V5| V2|  expertOf|
| V2| V4|correlated|
| V4| V2|correlated|
+---+---+----------+

+---+----------+--------+
| id|entityName|    name|
+---+----------+--------+
| V1|      user|   Paolo|
| V2|     topic|     SQL|
| V3|      user|   David|
| V4|     topic|Big Data|
| V5|      user|    John|
+---+----------+--------+
```

[5]: (None, None)

[6]:
```
filteredEDF = eDF.filter("linktype='follow'")
```

[ ]:
```
g = GraphFrame(vDF, filteredEDF)
```

[ ]:
```
resultDF = g.find("(userID)-[follow]->(topicID)")
```

[11]:
```
resultDF.show(), resultDF.printSchema()
```

```
+----------------+---------------+--------------------+
|          userID|         follow|             topicID|
+----------------+---------------+--------------------+
|{V1, user, Paolo}|{V1, V3, follow}|   {V3, user, David}|
|{V1, user, Paolo}|{V1, V4, follow}|{V4, topic, Big D…|
|{V3, user, David}|{V3, V2, follow}|    {V2, topic, SQL}|
|{V3, user, David}|{V3, V4, follow}|{V4, topic, Big D…|
+----------------+---------------+--------------------+

root
```

```
 |-- userID: struct (nullable = false)
 |    |-- id: string (nullable = true)
 |    |-- entityName: string (nullable = true)
 |    |-- name: string (nullable = true)
 |-- follow: struct (nullable = false)
 |    |-- src: string (nullable = true)
 |    |-- dst: string (nullable = true)
 |    |-- linktype: string (nullable = true)
 |-- topicID: struct (nullable = false)
 |    |-- id: string (nullable = true)
 |    |-- entityName: string (nullable = true)
 |    |-- name: string (nullable = true)
```

[11]: (None, None)

[12]:
```
#qui faccio un filter perchè un utente può anche followare un altro utente␣
 ↪oltre che un topic
topicsDF = resultDF.filter("userID.entityName='user' AND topicID.
 ↪entityName='topic'")
```

[13]:
```
topicsDF.show(), topicsDF.printSchema()
```

```
+----------------+---------------+--------------------+
|          userID|         follow|             topicID|
+----------------+---------------+--------------------+
|{V1, user, Paolo}|{V1, V4, follow}|{V4, topic, Big D…|
|{V3, user, David}|{V3, V2, follow}|    {V2, topic, SQL}|
|{V3, user, David}|{V3, V4, follow}|{V4, topic, Big D…|
+----------------+---------------+--------------------+

root
 |-- userID: struct (nullable = false)
 |    |-- id: string (nullable = true)
 |    |-- entityName: string (nullable = true)
 |    |-- name: string (nullable = true)
 |-- follow: struct (nullable = false)
 |    |-- src: string (nullable = true)
 |    |-- dst: string (nullable = true)
 |    |-- linktype: string (nullable = true)
 |-- topicID: struct (nullable = false)
 |    |-- id: string (nullable = true)
 |    |-- entityName: string (nullable = true)
 |    |-- name: string (nullable = true)
```

[13]: (None, None)

[14]:
```
finalDF = topicsDF.selectExpr("userID.name AS userName", "topicID.name AS␣
↪topicName")
```

[15]:
```
finalDF.show()
```

```
+--------+---------+
|userName|topicName|
+--------+---------+
|   Paolo| Big Data|
|   David|      SQL|
|   David| Big Data|
+--------+---------+
```

[16]:
```
finalDF.write.csv(outputPath, header=True)
```