STA323

Big Data Analysis Software and Application (Hadoop or Spark)

Report on Assignment 2

12112627 李乐平

Question 1. The protein sequence data is in the remote server /shareddata/data/as2/Q1_data. Here, fasta is a widely used file format in bioinformatics. In the fasta file, the line starting with '>' is the header of the sequence, i.e. comment for the protein sequence.

在题目的最后声明忽略氨基酸序列的拼接问题，但在此之前我已经实现了氨基酸序列的拼接，故实现此部分的注释和代码以加删除线的形式呈现。

(1). Calculate the frequency of each type of amino acid with Spark Dataframe API. [1 point]

Answer:

首先筛选出注释行（以'>'开头的行），用 is_header 列标记之~~，并用 is_header 的前缀和 group 标记氨基酸的分组（即其属于第几个蛋白质）~~。用 line_num 标记行号，然后删去注释行。

```
fasta_df = fasta_df.withColumn("is_header", when(col("value").startswith(">"), 1).otherwise(0))
fasta_df = fasta_df.withColumn("line_num", monotonically_increasing_id())
~~fasta_df = fasta_df.withColumn("group", expr("sum(int(is_header)) over (order by line_num)"))~~
filtered_df = fasta_df.filter(col("is_header") == 0)
```

```
+--------------------+---------+--------+-----+
|               value|is_header|line_num|group|
+--------------------+---------+--------+-----+
|MEEITQIKKRLSQTVRL...|        0|       1|    1|
|KKEVVAVAKKEEVLKKE...|        0|       2|    1|
|DIVPQMRDVSLPPKEEE...|        0|       3|    1|
|SLPPKKDEEIVCEKKEV...|        0|       4|    1|
|KEHEEKETFVVLKKEII...|        0|       5|    1|
+--------------------+---------+--------+-----+
```

~~可以用下列代码将同组的氨基酸序列拼接。~~

```
~~merged_df = filtered_df.groupBy("group").agg(concat_ws("", collect_list("value")).alias("sequence"))~~
```

使用 explode 函数展开各行并统计，即得各氨基酸出现的频数。

```
amino_acid_df = merged_df.select(explode(split(col("value"), "")).alias("amino_acid"))
frequency_df = amino_acid_df.groupBy("amino_acid").count().orderBy("amino_acid")
```

```
+----------+-------+
|amino_acid|  count|
+----------+-------+
|         A|3223081|
|         B|      6|
|         C| 564455|
|         D|2216904|
|         E|2674664|
|         F| 985877|
|         G|2653426|
|         H| 628384|
|         I|1726915|
|         K|1684031|
|         L|2851645|
|         M| 467474|
|         N|1316889|
|         P|2097950|
|         Q|1422769|
|         R|1789613|
|         S|2747798|
|         T|2795042|
|         V|2760761|
|         W| 351166|
|         X|   1347|
|         Y| 823096|
|         Z|      2|
+----------+-------+
```

(2). Complete the above work with Spark RDD API. [1 point]

Answer:

使用 RDD 实现如前所述的流程即可。

```
def is_header(line):
    return line.startswith(">")

def assign_group(iterator):
    group = 0
    for line in iterator:
        if is_header(line):
            group += 1
        yield (group, line)

grouped_rdd = fasta_rdd.mapPartitions(assign_group)
filtered_rdd = grouped_rdd.filter(lambda x: not is_header(x[1]))
merged_rdd = filtered_rdd.groupByKey().mapValues(lambda lines: "".join(lines))
amino_acid_rdd = merged_rdd.flatMap(lambda x: list(x[1]))
frequency_rdd = amino_acid_rdd.map(lambda x: (x, 1)).reduceByKey(lambda a, b: a + b).sortByKey()
```

```
Frequency of each amino acid:
A 3223081
B 6
C 564455
D 2216904
E 2674664
F 985877
G 2653426
H 628384
I 1726915
K 1684031
L 2851645
M 467474
N 1316889
P 2097950
Q 1422769
R 1789613
S 2747798
T 2795042
V 2760761
W 351166
X 1347
Y 823096
Z 2
```

(3). Count the number of a specific sequence motif "STAT" with Spark. [2 points]

Answer:

统计每行出现的 STAT 字样的次数并求和即可，得到共 2052 个。（如果考虑拼接则数量稍多一些）

```
stat_count_df = merged_df.withColumn("stat_occurrences", (size(split(col("value"), "STAT")) - 1))
stat_count_df.show(5)
total_stat_count = stat_count_df.agg({"stat_occurrences": "sum"}).collect()[0][0]
```

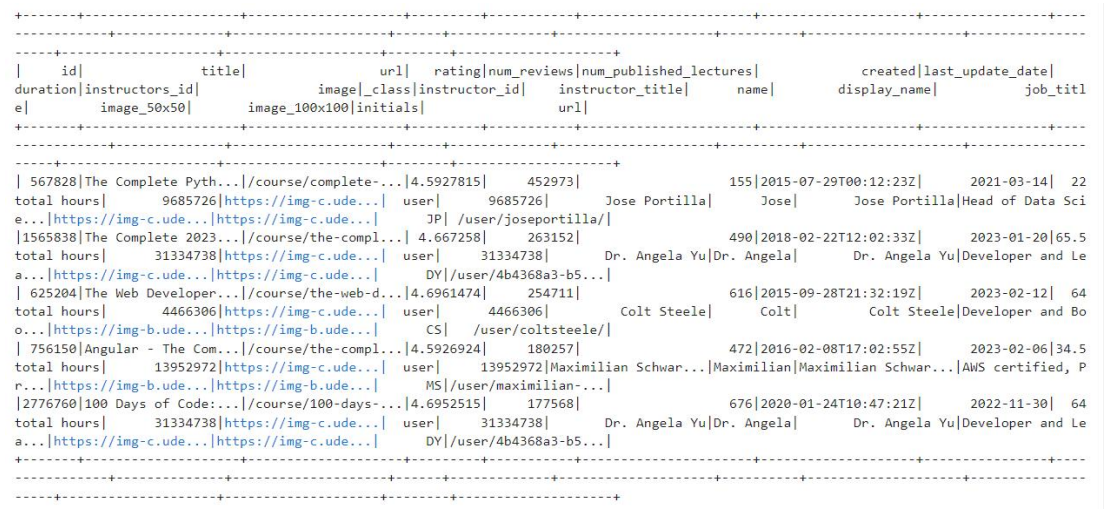
Question 2. Here is a dataset related to the online open courses. The course.csv contains the information about the courses while instructors.csv contains the information about the instructors.

(1). Join the 2 dataframes by instructors_id, use the inner join function. [2 points]

Answer:

```
instructor_df_renamed = instructor_df.withColumnRenamed("id",
"instructor_id").withColumnRenamed("title", "instructor_title")
joined_df = course_df.join(
    instructor_df_renamed,
    instructor_df_renamed.instructor_id == course_df.instructors_id, how = "inner"
)
joined_df.drop(joined_df["instructors_id"])
```

注意处理重名的列即可。



```
+-------+--------------------+--------------------+---------+-----------+----------------------+--------------------+----------------+----
-----------+--------------+--------------------+------+-------------+--------------------+----------+--------------------+--------------
-----+------------------+--------------------+--------------------+--------+--------------------+
|     id|               title|                 url|   rating|num_reviews|num_published_lectures|             created|last_update_date|
duration|instructors_id|               image|_class|instructor_id|    instructor_title|      name|        display_name|           job_titl
e|         image_50x50|       image_100x100|initials|                 url|
+-------+--------------------+--------------------+---------+-----------+----------------------+--------------------+----------------+----
-----------+--------------+--------------------+------+-------------+--------------------+----------+--------------------+--------------
-----+------------------+--------------------+--------------------+--------+--------------------+
| 567828|The Complete Pyth...|/course/complete-...|4.5927815|     452973|                   155|2015-07-29T00:12:23Z|      2021-03-14|  22
total hours|       9685726|https://img-c.ude...|  user|      9685726|       Jose Portilla|      Jose|       Jose Portilla|Head of Data Sci
e...|https://img-c.ude...|https://img-c.ude...|      JP| /user/joseportilla/|
|1565838|The Complete 2023...|/course/the-compl...| 4.667258|     263152|                   490|2018-02-22T12:02:33Z|      2023-01-20|65.5
total hours|      31334738|https://img-c.ude...|  user|     31334738|       Dr. Angela Yu|Dr. Angela|       Dr. Angela Yu|Developer and Le
a...|https://img-c.ude...|https://img-c.ude...|      DY|/user/4b4368a3-b5...|
| 625204|The Web Developer...|/course/the-web-d...|4.6961474|     254711|                   616|2015-09-28T21:32:19Z|      2023-02-12|  64
total hours|       4466306|https://img-c.ude...|  user|      4466306|         Colt Steele|      Colt|         Colt Steele|Developer and Bo
o...|https://img-b.ude...|https://img-b.ude...|      CS|   /user/coltsteele/|
| 756150|Angular - The Com...|/course/the-compl...|4.5926924|     180257|                   472|2016-02-08T17:02:55Z|      2023-02-06|34.5
total hours|      13952972|https://img-c.ude...|  user|     13952972|Maximilian Schwar...|Maximilian|Maximilian Schwar...|AWS certified, P
r...|https://img-b.ude...|https://img-b.ude...|      MS|/user/maximilian-...|
|2776760|100 Days of Code:...|/course/100-days-...|4.6952515|     177568|                   676|2020-01-24T10:47:21Z|      2022-11-30|  64
total hours|      31334738|https://img-c.ude...|  user|     31334738|       Dr. Angela Yu|Dr. Angela|       Dr. Angela Yu|Developer and Le
a...|https://img-c.ude...|https://img-c.ude...|      DY|/user/4b4368a3-b5...|
+-------+--------------------+--------------------+---------+-----------+----------------------+--------------------+----------------+----
-----------+--------------+--------------------+------+-------------+--------------------+----------+--------------------+--------------
-----+------------------+--------------------+--------------------+--------+--------------------+
```

(2). Use SQL (in pyspark SQL) to show the display_name and job_title of the instructor, who has the highest course rating among all courses that are related to "spark". [2 points]

Answer:

首先创建视图。

```
course_df.createOrReplaceTempView("course")
instructor_df.createOrReplaceTempView("instructor")
joined_df.createOrReplaceTempView("course_with_instructor")
```

依照题意筛选排序即可。

```
q2_2_df = spark.sql("""
    select
        display_name,
        job_title,
        rating
    from
        course_with_instructor
    where
        title like "%spark%"
    and
        created >= "2018-01-01 00:00:00"
    and
        rating = (
            select
                max(rating)
            from
                course_with_instructor
            where
                title like "%spark%"
            and
                created >= "2018-01-01 00:00:00"
        )
    order by
        rating desc
""")
q2_2_df.show(truncate = False)
```

```
+------------+--------------------------------------+---------+
|display_name|job_title                             |rating   |
+------------+--------------------------------------+---------+
|Deby Coles  |Sewer, Artist, Crafter and Instructor|4.6432705|
+------------+--------------------------------------+---------+
```

(3). Use SQL (in pyspark SQL) to select all courses that are contains "interview" or "interviews" and then sorted by course_rating in descending order and created in descending order (newest first). In this task, the course rating should be firstly rounded to one decimal place. [2 points]

Answer:

依照题意，注意将 title 小写化进行匹配，编写 SQL 即可。这里仅选中了必要的列。

```
q2_3_df = spark.sql("""
    select
        id,
        title,
        created,
        round(rating, 1) AS rating
    from
        course
    where
        lower(title) like "%interview%"
    order by
        rating desc,
        created desc
""")
q2_3_df.show(truncate = False)
```

```
+-------+-----------------------------------------------------------+--------------------+------+
|id     |title                                                      |created             |rating|
+-------+-----------------------------------------------------------+--------------------+------+
|4886926|Interview Oriented Data Structure Arrays & Linked List C|C++|2022-09-17T17:57:14Z|5.0   |
|4309400|CATIA V5 FOR JOBS INTERVIEW                                |2021-09-20T12:54:23Z|5.0   |
|4829150|Réaliser des interviews au rendu professionnel (PARTIE 2)  |2022-08-12T14:54:06Z|4.9   |
|4722894|"The ""BigTech"" System Design Interview Bootcamp"          |2022-06-07T14:53:40Z|4.9   |
|4499476|Power BI Interview Questions and Answers                   |2022-01-17T11:08:03Z|4.9   |
```