Big Data Frameworks CSE3120

Experiment: User Defined Function using Scala in spark framework

Name: Naveen Nidadavolu
Roll No: 22MIA1049

Aim:

To implement a **User Defined Function (UDF) using Scala** in the **Apache Spark framework**, which transforms the input data by capitalizing the first letter of each word in a given column.

Algorithm:

1. **Start**
2. **Initialize Spark Session**:
   - Import necessary libraries.
   - Create a Spark session.
3. **Define Column Names**:
   - Create a sequence of column names.
4. **Create Sample Data**:
   - Define a sequence of tuples containing sample data.
5. **Convert Data into DataFrame**:
   - Use toDF() to convert the sequence into a Spark DataFrame.
6. **Define a User Defined Function (UDF)**:
   - Create a function that takes a string, splits it into words, capitalizes the first letter of each word, and then rejoins them.
7. **Register the UDF in Spark**:
   - Convert the function into a UDF using udf().
8. **Apply the UDF to the DataFrame**:
   - Use the select() method to apply the UDF to the specific column.
   - Rename the transformed column appropriately.
9. **Display the Output**:
   - Show the DataFrame with the transformed data.
10. **End**

Procedure

Step 1: Start Spark Shell

- Open the terminal and start the Spark shell.
- Ensure Spark is running successfully.

```
25/03/24 11:08:07 WARN NativeCodeLoader: Unable to load native-hadoop library fo
r your platform... using builtin-java classes where applicable
25/03/24 11:08:07 WARN Utils: Your hostname, ubuntu resolves to a loopback addre
ss: 127.0.1.1; using 10.0.2.15 instead (on interface eth0)
25/03/24 11:08:07 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another
address
25/03/24 11:08:10 WARN ObjectStore: Failed to get database global_temp, returnin
g NoSuchObjectException
Spark context Web UI available at http://10.0.2.15:4040
Spark context available as 'sc' (master = local[*], app id = local-1742794687968
).
Spark session available as 'spark'.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 2.2.0
      /_/

Using Scala version 2.11.8 (Java HotSpot(TM) Client VM, Java 1.8.0_45)
Type in expressions to have them evaluated.
Type :help for more information.

scala>
```

Step 2: Define Column Names

- Define a sequence of column names.

```
scala> import spark.implicits._
import spark.implicits._

scala> val cols = Seq("sno","name")
cols: Seq[String] = List(sno, name)
```

- This sequence will be used for naming the DataFrame columns.

Step 3: Create Sample Data

- Define a sequence of tuples containing sample records.
- Convert it into a DataFrame with the specified column names.
- The DataFrame is displayed with `sno` and `name` columns.

```
scala> val data = Seq(("1","naveen"),
     | ("2","chernoAlpha"),
     | ("3","crimpsonTyphoon")
     | )
data: Seq[(String, String)] = List((1,naveen), (2,chernoAlpha), (3,crimpsonTyphoon))

scala> val df = data.toDF(cols:_*)
df: org.apache.spark.sql.DataFrame = [sno: string, name: string]

scala> df.show(false)
+---+---------------+
|sno|name           |
+---+---------------+
|1  |naveen         |
|2  |chernoAlpha    |
|3  |crimpsonTyphoon|
+---+---------------+


scala>
```

Step 4: Define User Defined Function (UDF)

- Define a function to capitalize the first letter of each word in the "name" column.
- Convert the function into a Spark User Defined Function (UDF).

```
scala> val Ucase = (strQuote:String) => {
     | val dt = strQuote.split(" ")
     | dt.map(f => f.substring(0,1).toUpperCase + f.substring(1,f.length)).mkString(
" ")
     | }
Ucase: String => String = <function1>

scala> val customUDF = udf(Ucase)
customUDF: org.apache.spark.sql.expressions.UserDefinedFunction = UserDefinedFunctio
n(<function1>,StringType,Some(List(StringType)))
```

Step 5: Apply UDF on DataFrame

- Apply the UDF to modify the "name" column and display the updated DataFrame.

```
scala> df.select(col("sno"), customUDF(col("name")).as("name")).show(false)
+---+---------------+
|sno|name           |
+---+---------------+
|1  |Naveen         |
|2  |ChernoAlpha    |
|3  |CrimpsonTyphoon|
+---+---------------+
```

Result:

The Spark User Defined Function (UDF) successfully transformed the "name" column by capitalizing the first letter of each word. The updated DataFrame displayed the modified names while retaining the "sno" column values.