Spark SQL Agent

This notebook shows how to use agents to interact with a Spark SQL. Similar to SQL Database Agent, it is designed to address general inquiries about Spark SQL and facilitate error recovery.

NOTE: Note that, as this agent is in active development, all answers might not be correct. Additionally, it is not guaranteed that the agent won't perform DML statements on your Spark cluster given certain questions. Be careful running it on sensitive data!

Initialization

```
from langchain.agents import create_spark_sql_agent
from langchain.agents.agent_toolkits import SparkSQLToolkit
from langchain.chat_models import ChatOpenAI
from langchain.utilities.spark_sql import SparkSQL
```

```
from pyspark.sql import SparkSession

spark = SparkSession.builder.getOrCreate()
schema = "langchain_example"
spark.sql(f"CREATE DATABASE IF NOT EXISTS {schema}")
spark.sql(f"USE {schema}")
csv_file_path = "titanic.csv"
table = "titanic"
```

```
spark.read.csv(csv_file_path, header=True, inferSchema=True).write.saveAsTable(table)
spark.table(table).show()
```

```
    Setting default log level to "WARN".
    To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
    23/05/18 16:03:10 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using
builtin-java classes where applicable


    +-----------+--------+------+--------------------+------+----+-----+-----+----------------+-------+-----
+--------+
    |PassengerId|Survived|Pclass|                Name|   Sex| Age|SibSp|Parch|          Ticket|
Fare|Cabin|Embarked|
    +-----------+--------+------+--------------------+------+----+-----+-----+----------------+-------+-----
+--------+
    |          1|       0|     3|Braund, Mr. Owen ...|  male|22.0|    1|    0|       A/5 21171|   7.25| null|
S|
    |          2|       1|     1|Cumings, Mrs. Joh...|female|38.0|    1|    0|        PC 17599|71.2833|  C85|
C|
    |          3|       1|     3|Heikkinen, Miss. ...|female|26.0|    0|    0|STON/O2. 3101282|  7.925| null|
S|
    |          4|       1|     1|Futrelle, Mrs. Ja...|female|35.0|    1|    0|          113803|   53.1| C123|
S|
    |          5|       0|     3|Allen, Mr. Willia...|  male|35.0|    0|    0|          373450|   8.05| null|
S|
    |          6|       0|     3|    Moran, Mr. James|  male|null|    0|    0|          330877| 8.4583| null|
Q|
    |          7|       0|     1|McCarthy, Mr. Tim...|  male|54.0|    0|    0|           17463|51.8625|  E46|
S|
    |          8|       0|     3|Palsson, Master. ...|  male| 2.0|    3|    1|          349909| 21.075| null|
S|
    |          9|       1|     3|Johnson, Mrs. Osc...|female|27.0|    0|    2|          347742|11.1333| null|
```

```
S|
    |         10|       1|     2|Nasser, Mrs. Nich...|female|14.0|    1|    0|          237736|30.0708| null|
C|
    |         11|       1|     3|Sandstrom, Miss. ...|female| 4.0|    1|    1|         PP 9549|   16.7|   G6|
S|
    |         12|       1|     1|Bonnell, Miss. El...|female|58.0|    0|    0|          113783|  26.55| C103|
S|
    |         13|       0|     3|Saundercock, Mr. ...|  male|20.0|    0|    0|       A/5. 2151|   8.05| null|
S|
    |         14|       0|     3|Andersson, Mr. An...|  male|39.0|    1|    5|          347082| 31.275| null|
S|
    |         15|       0|     3|Vestrom, Miss. Hu...|female|14.0|    0|    0|          350406| 7.8542| null|
S|
    |         16|       1|     2|Hewlett, Mrs. (Ma...|female|55.0|    0|    0|          248706|   16.0| null|
S|
    |         17|       0|     3|Rice, Master. Eugene|  male| 2.0|    4|    1|          382652| 29.125| null|
Q|
    |         18|       1|     2|Williams, Mr. Cha...|  male|null|    0|    0|          244373|   13.0| null|
S|
    |         19|       0|     3|Vander Planke, Mr...|female|31.0|    1|    0|          345763|   18.0| null|
S|
    |         20|       1|     3|Masselmani, Mrs. ...|female|null|    0|    0|            2649|  7.225| null|
C|
    +-----------+--------+------+--------------------+------+----+-----+-----+----------------+-------+-----
+--------+
    only showing top 20 rows
```

```
# Note, you can also connect to Spark via Spark connect. For example:
# db = SparkSQL.from_uri("sc://localhost:15002", schema=schema)
spark_sql = SparkSQL(schema=schema)
llm = ChatOpenAI(temperature=0)
```

```
toolkit = SparkSQLToolkit(db=spark_sql, llm=llm)
agent_executor = create_spark_sql_agent(llm=llm, toolkit=toolkit, verbose=True)
```

Example: describing a table

```
agent_executor.run("Describe the titanic table")
```

```
> Entering new AgentExecutor chain...
Action: list_tables_sql_db
Action Input:
Observation: titanic
Thought:I found the titanic table. Now I need to get the schema and sample rows for the titanic table.
Action: schema_sql_db
Action Input: titanic
Observation: CREATE TABLE langchain_example.titanic (
  PassengerId INT,
  Survived INT,
  Pclass INT,
  Name STRING,
  Sex STRING,
  Age DOUBLE,
  SibSp INT,
  Parch INT,
  Ticket STRING,
  Fare DOUBLE,
  Cabin STRING,
```

```
      Embarked STRING)
    ;

    /*
    3 rows from titanic table:
    PassengerId Survived    Pclass  Name    Sex Age SibSp   Parch   Ticket  Fare    Cabin   Embarked
    1   0   3   Braund, Mr. Owen Harris male    22.0    1   0   A/5 21171   7.25    None    S
    2   1   1   Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38.0    1   0   PC 17599
71.2833 C85 C
    3   1   3   Heikkinen, Miss. Laina  female  26.0    0   0   STON/O2. 3101282    7.925   None    S
    */
    Thought:I now know the schema and sample rows for the titanic table.
    Final Answer: The titanic table has the following columns: PassengerId (INT), Survived (INT), Pclass
(INT), Name (STRING), Sex (STRING), Age (DOUBLE), SibSp (INT), Parch (INT), Ticket (STRING), Fare (DOUBLE),
Cabin (STRING), and Embarked (STRING). Here are some sample rows from the table:

    1. PassengerId: 1, Survived: 0, Pclass: 3, Name: Braund, Mr. Owen Harris, Sex: male, Age: 22.0, SibSp: 1,
Parch: 0, Ticket: A/5 21171, Fare: 7.25, Cabin: None, Embarked: S
    2. PassengerId: 2, Survived: 1, Pclass: 1, Name: Cumings, Mrs. John Bradley (Florence Briggs Thayer),
Sex: female, Age: 38.0, SibSp: 1, Parch: 0, Ticket: PC 17599, Fare: 71.2833, Cabin: C85, Embarked: C
    3. PassengerId: 3, Survived: 1, Pclass: 3, Name: Heikkinen, Miss. Laina, Sex: female, Age: 26.0, SibSp:
0, Parch: 0, Ticket: STON/O2. 3101282, Fare: 7.925, Cabin: None, Embarked: S

    > Finished chain.
```

```
    'The titanic table has the following columns: PassengerId (INT), Survived (INT), Pclass (INT), Name
(STRING), Sex (STRING), Age (DOUBLE), SibSp (INT), Parch (INT), Ticket (STRING), Fare (DOUBLE), Cabin
(STRING), and Embarked (STRING). Here are some sample rows from the table: \n\n1. PassengerId: 1, Survived:
0, Pclass: 3, Name: Braund, Mr. Owen Harris, Sex: male, Age: 22.0, SibSp: 1, Parch: 0, Ticket: A/5 21171,
```

```
Fare: 7.25, Cabin: None, Embarked: S\n2. PassengerId: 2, Survived: 1, Pclass: 1, Name: Cumings, Mrs. John Bradley (Florence Briggs Thayer), Sex: female, Age: 38.0, SibSp: 1, Parch: 0, Ticket: PC 17599, Fare: 71.2833, Cabin: C85, Embarked: C\n3. PassengerId: 3, Survived: 1, Pclass: 3, Name: Heikkinen, Miss. Laina, Sex: female, Age: 26.0, SibSp: 0, Parch: 0, Ticket: STON/O2. 3101282, Fare: 7.925, Cabin: None, Embarked: S'
```

Example: running queries

```
agent_executor.run("whats the square root of the average age?")
```

```
> Entering new AgentExecutor chain...
Action: list_tables_sql_db
Action Input:
Observation: titanic
Thought:I should check the schema of the titanic table to see if there is an age column.
Action: schema_sql_db
Action Input: titanic
Observation: CREATE TABLE langchain_example.titanic (
  PassengerId INT,
  Survived INT,
  Pclass INT,
  Name STRING,
  Sex STRING,
  Age DOUBLE,
  SibSp INT,
  Parch INT,
  Ticket STRING,
```

```
      Fare DOUBLE,
      Cabin STRING,
      Embarked STRING)
    ;

    /*
    3 rows from titanic table:
    PassengerId Survived    Pclass  Name    Sex Age SibSp   Parch   Ticket  Fare    Cabin   Embarked
    1   0   3   Braund, Mr. Owen Harris male    22.0    1   0   A/5 21171   7.25    None    S
    2   1   1   Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38.0    1   0   PC 17599
71.2833 C85 C
    3   1   3   Heikkinen, Miss. Laina  female  26.0    0   0   STON/O2. 3101282    7.925   None    S
    */
    Thought:There is an Age column in the titanic table. I should write a query to calculate the average age
and then find the square root of the result.
    Action: query_checker_sql_db
    Action Input: SELECT SQRT(AVG(Age)) as square_root_of_avg_age FROM titanic
    Observation: The original query seems to be correct. Here it is again:

    SELECT SQRT(AVG(Age)) as square_root_of_avg_age FROM titanic
    Thought:The query is correct, so I can execute it to find the square root of the average age.
    Action: query_sql_db
    Action Input: SELECT SQRT(AVG(Age)) as square_root_of_avg_age FROM titanic
    Observation: [('5.449689683556195',)]
    Thought:I now know the final answer
    Final Answer: The square root of the average age is approximately 5.45.

    > Finished chain.
```

```
    'The square root of the average age is approximately 5.45.'
```

```
agent_executor.run("What's the name of the oldest survived passenger?")
```

```
    > Entering new AgentExecutor chain...
    Action: list_tables_sql_db
    Action Input:
    Observation: titanic
    Thought:I should check the schema of the titanic table to see what columns are available.
    Action: schema_sql_db
    Action Input: titanic
    Observation: CREATE TABLE langchain_example.titanic (
      PassengerId INT,
      Survived INT,
      Pclass INT,
      Name STRING,
      Sex STRING,
      Age DOUBLE,
      SibSp INT,
      Parch INT,
      Ticket STRING,
      Fare DOUBLE,
      Cabin STRING,
      Embarked STRING)
    ;

    /*
    3 rows from titanic table:
```

```
    PassengerId Survived    Pclass  Name    Sex Age SibSp   Parch   Ticket  Fare    Cabin   Embarked
    1   0   3   Braund, Mr. Owen Harris male    22.0    1   0   A/5 21171   7.25    None    S
    2   1   1   Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38.0    1   0   PC 17599
71.2833 C85 C
    3   1   3   Heikkinen, Miss. Laina  female  26.0    0   0   STON/O2. 3101282    7.925   None    S
    */
    Thought:I can use the titanic table to find the oldest survived passenger. I will query the Name and Age
columns, filtering by Survived and ordering by Age in descending order.
    Action: query_checker_sql_db
    Action Input: SELECT Name, Age FROM titanic WHERE Survived = 1 ORDER BY Age DESC LIMIT 1
    Observation: SELECT Name, Age FROM titanic WHERE Survived = 1 ORDER BY Age DESC LIMIT 1
    Thought:The query is correct. Now I will execute it to find the oldest survived passenger.
    Action: query_sql_db
    Action Input: SELECT Name, Age FROM titanic WHERE Survived = 1 ORDER BY Age DESC LIMIT 1
    Observation: [('Barkworth, Mr. Algernon Henry Wilson', '80.0')]
    Thought:I now know the final answer.
    Final Answer: The oldest survived passenger is Barkworth, Mr. Algernon Henry Wilson, who was 80 years
old.

    > Finished chain.



    'The oldest survived passenger is Barkworth, Mr. Algernon Henry Wilson, who was 80 years old.'
```